Yngrith Soares da Silva

Redução de potência e interferência através das metaheurísticas GRASP e VNS em redes de sensores sem fio

Vitória, ES

2020

Yngrith Soares da Silva

Redução de potência e interferência através das metaheurísticas GRASP e VNS em redes de sensores sem fio

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Mestre em Engenharia Elétrica.

Universidade Federal do Espírito Santo – UFES
Centro Tecnológico
Programa de Pós-Graduação em Engenharia Elétrica

Orientador: Prof. Dr. Helder Roberto de O. Rocha
Coorientador: Prof. Dr. Jair Adriano Lima Silva

Vitória, ES
2020

Ficha catalográfica disponibilizada pelo Sistema Integrado de
Bibliotecas - SIBI/UFES e elaborada pelo autor

S586r

Silva, Yngrith Soares da, 1989-
Redução de potência e interferência através das metaheurísticas GRASP e VNS em redes de sensores sem fio / Yngrith Soares da Silva. - 2020.
112 f. : il.

Orientador: Helder Roberto de Oliveira Rocha.
Coorientador: Jair Adriano Lima Silva.
Dissertação (Mestrado em Engenharia Elétrica) - Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Sistemas de comunicação sem fio. I. Rocha, Helder Roberto de Oliveira. II. Silva, Jair Adriano Lima. III. Universidade Federal do Espírito Santo. Centro Tecnológico. IV. Título.

CDU: 621.3

Yngrith Soares da Silva

Redução de potência e interferência através das metaheurísticas GRASP e VNS em redes de sensores sem fio

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Mestre em Engenharia Elétrica.

Trabalho aprovado. Vitória, ES, 18 de dezembro de 2020:

Prof. Dr. Helder Roberto de O. Rocha
Orientador

Professor: Jair Adriano Lima Silva
Coorientador

Professor: Marcia Helena M. Paiva
Examinadora Interna

Professor: Wanderley Cardoso Celeste
Examinador Externo

Vitória, ES
2020

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

PROTOCOLO DE ASSINATURA

O documento acima foi assinado digitalmente com senha eletrônica através do Protocolo Web, conforme Portaria UFES nº 1.269 de 30/08/2018, por
WANDERLEY CARDOSO CELESTE - SIAPE 1723581
Departamento de Computação e Eletrônica - DCE/CEUNES
Em 27/04/2021 às 11:10

Para verificar as assinaturas e visualizar o documento original acesse o link:
https://api.lepisma.ufes.br/arquivos-assinados/179371?tipoArquivo=O

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

PROTOCOLO DE ASSINATURA

O documento acima foi assinado digitalmente com senha eletrônica através do Protocolo Web, conforme Portaria UFES nº 1.269 de 30/08/2018, por
MARCIA HELENA MOREIRA PAIVA - SIAPE 2524599
Departamento de Engenharia Elétrica - DEE/CT
Em 29/04/2021 às 11:11

Para verificar as assinaturas e visualizar o documento original acesse o link:
https://api.lepisma.ufes.br/arquivos-assinados/180684?tipoArquivo=O

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

PROTOCOLO DE ASSINATURA

O documento acima foi assinado digitalmente com senha eletrônica através do Protocolo Web, conforme Portaria UFES nº 1.269 de 30/08/2018, por
HELDER ROBERTO DE OLIVEIRA ROCHA - SIAPE 1860639
Departamento de Engenharia Elétrica - DEE/CT
Em 29/04/2021 às 14:28

Para verificar as assinaturas e visualizar o documento original acesse o link:
https://api.lepisma.ufes.br/arquivos-assinados/180824?tipoArquivo=O

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

PROTOCOLO DE ASSINATURA

O documento acima foi assinado digitalmente com senha eletrônica através do Protocolo Web, conforme Portaria UFES nº 1.269 de 30/08/2018, por
JAIR ADRIANO LIMA SILVA - SIAPE 1514954
Departamento de Engenharia Elétrica - DEE/CT
Em 14/05/2021 às 10:17

Para verificar as assinaturas e visualizar o documento original acesse o link:
https://api.lepisma.ufes.br/arquivos-assinados/189831?tipoArquivo=O

Para Vinícios, João, Jéssica e Davi, razões de minha vida.
Para Luis Carlos e Sirlene, que me deram a vida.

Agradecimentos

Aprendi ao longo destes anos, no Mestrado, que não conquistamos nada sozinho. Sempre necessitamos de carinho, apoio e palavras de incentivo. Por isso, quero agradecer a quem me acompanhou neste caminho.

Primeiramente, Senhor, obrigada por mais essa vitória que me deu! Obrigada por me fazer querer lutar nas horas mais difíceis. Agradeço imensamente a minha mãe Sirlene, por me amar incondicionalmente e ter lutado para que eu tivesse mais essa conquista. Agradeço ao meu pai Luis Carlos (in memoriam), pois sempre esteve ao meu lado lutando e possibilitando a realização dos meus sonhos.

Agradeço aos meus irmãos, Jéssica e Davi, que sempre me deram todo o carinho necessário. Agradeço ao meu esposo, Marcos Vinícios, que sempre me apoiou e me incentivou. Agradeço ao meu filho, João Vitor, pois sem o seu sorriso essa batalha seria mais difícil.

Agradeço aos meus familiares e aos familiares do meu esposo, pois foram escolhidos por Deus para estarem ao meu lado.

Agradeço aos meus professores e orientadores que me ajudaram e apoiaram para a conclusão deste Mestrado, em especial: Helder Rocha, Renato Elias e Jair Silva. Agradeço ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) pela bolsa de estudos, que possibilitou a dedicação integral ao programa de pós-graduação e a operacionalização do estudo. Agradeço, também, ao Labtel e ao PPGEE.

Muito obrigada, pois sem vocês nada disso seria possível!

"Não sabendo que era impossível, foi lá e fez." (Jean Cocteau)

Resumo

Uma rede de sensores sem fio (RSSF) é formada por nós sensores que transmitem informação em canal na qual ocorrem transmissões multipercurso. Por este motivo podem ocorrer transmissões simultâneas, que ocasionam interferências provocando desvanecimento de potência. Um problema enfrentado pelas RSSF é a conservação da energia, que é utilizada para comunicação entre os nós. Geralmente, o fornecimento de energia é feito por baterias, que não podem ter grandes dimensões, não podem ser trocadas com facilidade e possuem alto custo de manutenção. Para melhor aproveitamento da energia disponível, pode-se utilizar uma técnica conhecida como Controle de Topologia. O Controle de Topologia determina os links de transmissão entre os nós sensores, mantendo a rede conexa. Desta forma, visa obter uma comunicação eficiente através da atribuição de potência de transmissão dos nós sensores, consequentemente reduzindo a interferência e também as retransmissões.

No presente trabalho, são propostas as metaheurísticas GRASP, VNS e GRASP-VNS para cinco variações do problema de atribuição de potência em RSSF (minimização de potência, minimização de interferência máxima, minimização de interferência total, minimização de potência e interferência máxima, minimização de potência e interferência total), garantindo que o grafo seja conexo. Além disso, os resultados obtidos nestas metaheurísticas foram comparados com resultados ótimos encontrados pelo CPLEX para as mesmas cinco variações do problema de atribuição de potência. As metaheurísticas propostas encontraram a mesma solução que os resultados ótimos para a maioria das instâncias pequenas (10 a 50 nós). Para instâncias realísticas (maiores que 100 nós), a metaheurística GRASP-VNS se mostrou mais promissora.

Palavras-chaves: Redes de sensores sem fio. Controle de topologia. Otimização de potência e interferência. Metaheurística. *Greedy Randomized Adaptive Search Procedure* (GRASP). *Variable Neighborhood Search* (VNS).

Abstract

A wireless sensor network (WSN) is formed by sensor nodes that transmit information on a channel in which multipath transmissions occur. For this reason, simultaneous transmissions can occur, which cause interference causing power to fade. A problem faced by the WSN is the conservation of energy, which is used for communication between nodes. Generally, the power supply is made by batteries, which cannot be large, cannot be easily exchanged and have a high maintenance cost. For better use of available energy, a technique known as Topology Control can be used. Topology Control determines the transmission links between sensor nodes, keeping the network connected. In this way, it aims to obtain an efficient communication through the attribution of transmission power of the sensor nodes, consequently reducing the interference and also the retransmissions.

In the present work, the GRASP, VNS and GRASP-VNS metaheuristics are proposed for five variations of the RSSF power allocation problem (minimization of power, minimization of maximum interference, minimization of total interference, minimization of power and maximum interference, minimization of power and total interference), ensuring that the graph is connected. In addition, the results obtained in these metaheuristics were compared with optimal results found by CPLEX for the same five variations of the power allocation problem. The proposed metaheuristics found the same solution as the optimal results for most small instances (10 to 50 knots). For realistic instances (greater than 100 knots), the GRASP-VNS metaheuristic proved to be more promising.

Keywords: Wireless sensor networks. Topology control. Optimization of power and interference. Metaheuristics. *Greedy Randomized Adaptive Search Procedure* (GRASP). *Variable Neighborhood Search* (VNS).

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

MANET	Mobile Ad Hoc Network
RSSF	Redes de sensores sem fio
PLIM	Programação Linear Inteira Mista
MPT	Minimização da Potência Total
MIMAX	Minimização de interferência máxima
MITOT	Minimização de interferência total
MIMAXP	Minimização de interferência máxima e potência
MITOTP	Minimização de interferência total e potência
RIM	*Receiver Interference Model*
SIM	*Sender Interference Model*
Dif	Diferença
GRASP	*Greedy Randomized Adaptive Search*
VNS	*Variable Neighborhood Search*
Qtde	Quantidade
Pot. tot.	Potência total
Pot	Potência
IM	Interferência Máxima
IT	Interferência Total
CC	Corrente Contínua
FI/SMEMIT	Fast Improved Strong Minimum Energy Minimum Interference Topology

Sumário

1 Introdução

Neste capítulo aborda-se alguns conceitos básicos necessários a compreensão desta Dissertação. Os conceitos apresentados estão organizados da seguinte forma: na Seção 1.1 é apresentado um resumo sobre os principais conceitos de grafos utilizados neste trabalho; na Seção 1.2 são abordados alguns conceitos básicos de RSSF; a motivação do trabalho é apresentada na seção 1.3; na seção 1.4 são apresentaods os objetivos da Dissertação e na seção 1.5 são apresentados os trabalhos da literatura que possuem semelhanças com a Dissertação.

1.1 Conceitos de Grafos

Tratar o problema de consumo de energia em redes de sensores sem fio (RSSF) pode se tornar uma tarefa difícil, mas as análises podem ser facilitadas se a rede é modelada através de Teoria dos grafos. Um grafo é formado por um conjunto não vazio de vértices V e um conjunto discreto de arestas E, de forma que: $G = (V, E)$ (BOAVENTURA-NETTO, 2012). Neste trabalho, as redes de sensores sem fio são representadas por grafos, sendo que os vértices representam os nós sensores e as arestas representam os enlaces de comunicação entre eles, como mostrado na Figura 1. Na esquerda, tem-se uma rede de sensores sem fio, na qual há nós sensores que possuem uma potência de transmissão, representada pelos círculos pontilhados. O nível de potência do nó determina com quais outros nós ele pode se comunicar. Esse fenômeno físico é representado por um grafo, mostrado à direita.

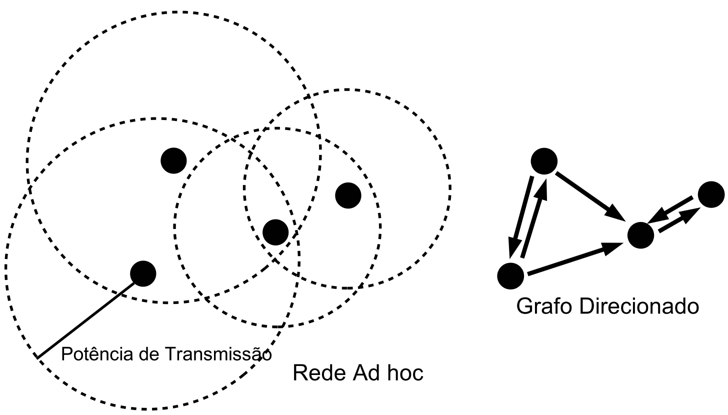


Figura 1 – Exemplo de uma RSSF (esquerda) representada por um grafo de comunicação (direita). Os vértices do grafo representam os nós sensores e os arcos, os *links* de comunicação entre eles.

Fonte: (MORAES; RIBEIRO; DUHAMEL, 2009)

A seguir são apresentadas algumas definições de Grafos utilizadas nesse trabalho:

- Um grafo é completo se for um grafo simples (não possuir laços e arestas paralelas) na qual todo par de vértices é ligado por uma aresta.

- As arestas representam a ligação entre os vértices e podem possuir peso. Uma aresta $\{i, j\}$ é a união de dois arcos unidirecionais ordenados $(i, j) \neq (j, i)$ tal que (i, j) significa $i \to j$ e (j, i) significa $j \to i$. Assim $\{i, j\} = (j, i) \cup (i, j)$, ou seja $\{i, j\} \Rightarrow i \leftrightarrow j$.

- O peso P é uma medida que pode ser fornecida a cada aresta. Em uma RSSF, o peso pode estar relacionado a distância entre os nós sensores, por exemplo.

- Um grafo $G' = (V', E')$ é subgrafo de um grafo $G = (V, E)$ se $V' \subseteq V$ e $E' \subseteq E$.

- Um grafo é dito valorado quando as arestas possuem valor associado a elas. Formalmente, é um grafo $G = (V, E)$, na qual há uma função f de E para P.

- Um grafo é direcionado quando as ligações entre os vértices são representadas por arcos. Eles modelam a RSSF como um grafo completo não direcionado $G = (V; E; w)$, onde V é um conjunto de vértices, E é um conjunto de arestas, onde cada aresta conecta dois nós e w é uma função de custo definido como $w(u; v) = d(u; v)$, onde $d(u; v)$ denota a distância euclidiana entre u e v.

- O grau de um vértice é representado pela quantidade de arestas incidentes no vértice quando o grafo é não-direcionado. Quando o grafo é direcionado, há o grau de entrada, que é representado pelo número de arcos que chegam em um vértice.

- No grafo direcionado, o grau de saída é representado pelo número de arcos que saem de um vétice.

- Um grafo é conexo se existe pelo menos um caminho entre cada par de vértices.

- Um grafo de comunicação $G = (V, E)$ é k-vértice-conexo se para dois vértices quaisquer u e v, existem k caminhos que não possuem vértices em comum conectando u e v, a não ser os próprios vértices u e v, ou seja, caminhos disjuntos por vértices.

- Um grafo de comunicação $G = (V, E)$ é k-aresta-conexo se para dois vértices quaisquer u e v, existem k caminhos que não possuem arestas em comum conectando u e v.

- Interferência de um nó u é definida como o número de nós sensores que ele alcança quando está se comunicando com um nó v da rede, exceto o nó v.

1.2 Conceitos de RSSF

Avanços nas áreas de microprocessadores, sensoriamento e comunicação sem fio levaram a utilização de redes de sensores em segurança, saúde, construção civil e automação industrial (LOUREIRO et al., 2003), por exemplo. Redes de sensores sem fio (RSSF) são formadas por centenas ou milhares de nós sensores distribuídos em uma área determinada e interconectados por uma tecnologia de comunicação sem fio (NIAR; HAFFAF, 2012). Cada nó pode adquirir dados relativos ao estado das variáveis do ambiente monitorado, como, por exemplo, umidade, ruído, luminosidade, calor, temperatura e pressão. Os nós sensores são, geralmente, formados por memória, processador, sensor e rádio comunicador.

As redes sem fio podem ser classificadas em: redes com infraestrutura e redes sem infraestrutura. A principal diferença entre as redes está relacionada com a forma de comunicação. Nas redes com infraestrutura, os dispositivos trocam informação utilizando estações-base, enquanto que nas redes sem infraestrutura a comunicação é feita diretamente de nó para nó. Redes sem infraestrutura também são conhecidas como redes *ad hoc*. Diversos autores classificam as RSSF como uma variante das redes *ad hoc*, conhecidas também como *Mobile Ad Hoc Network* (MANET) (LOUREIRO et al., 2003). Como não contam com uma infraestrutura pré-estabelecida, as RSSF enfrentam diversas restrições de energia e processamento.

Diferentemente das redes com infraestrutura, o fornecimento de energia para os nós, nas RSSF, é feito por baterias. Na maioria dos casos, as baterias têm peso e dimensões reduzidas, o que dificulta a instalação de baterias de longa duração. Há ainda o fato de que as baterias podem ser de difícil substituição como, por exemplo, em uma RSSF instalada em um ambiente hostil (fundo do mar ou boca de um vulcão). Por este motivo, um dos principais desafios das RSSF é construir algoritmos e protocolos de comunicação que ajudem a aumentar a vida útil das baterias, aumentando assim a longevidade da rede.

Devido a ausência de estações-base, em uma RSSF, os elementos sensores devem trocar dados através de seus rádios comunicadores. De acordo com Xing et al. [2005], a maior parte da energia consumida nas RSSF ocorre nos comunicadores. A comunicação pode ser realizada diretamente entre os nós ou através de um caminho com múltiplos nós. A comunicação direta entre os nós necessita de potências altas para que o nó origem alcance o destino diretamente. Como utilizam potências altas para transmitir dados diretamente, os nós das RSSFs estão sujeitos a interferências causadas por transmissões simultâneas. Se a comunicação entre os nós não ocorrer diretamente, mas através de um caminho com múltiplos nós, a potência total necessária para comunicação é inferior à potência da transmissão direta, reduzindo o consumo de energia. A Figura 2 mostra um exemplo de uma rede onde um nó pode se comunicar diretamente com outro (esquerda), mas estes nós podem se comunicar através de um caminho percorrendo múltiplos nós (direita). Percebe-se que a comunicação direta consumirá mais energia do nó transmissor do que as pequenas comunicações indiretas.

A Figura 3 mostra um exemplo de uma rede onde o nó D pode se comunicar diretamente com o nó E, mas estes nós podem se comunicar através de um caminho

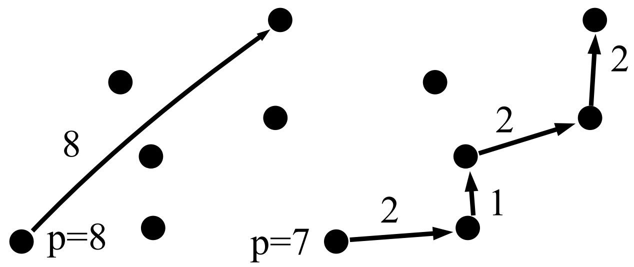


Figura 2 – Exemplo de comunicação direta entre nós de uma rede e através de caminho com diversos nós

percorrendo os nós D, C e E.

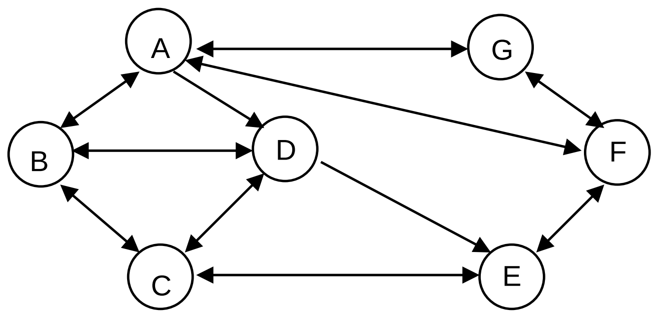


Figura 3 – Exemplo de comunicação direta e através de caminho com diversos nós em uma rede

O nó C está se comunicando com o nó D. Suponha que o nó D esteja recebendo uma mensagem do nó A no mesmo instante que C e E se comunicam. O nó D, por estar conectado aos nós C e E, sofrerá interferência, corrompendo a mensagem que o nó D iria receber do nó A, aumentando o consumo de energia para um novo envio da informação de A para D (TOH, 2001).

Neste trabalho, o objetivo é reduzir o consumo de energia da rede pelos comunicadores, seja através da redução de potência (transmissões com múltiplos caminhos de curta distância e baixa potência) ou pela redução de interferência (diminuição das retransmissões de mensagens corrompidas) ou de ambos.

1.3 Motivação

Um dos principais desafios relacionados às redes de sensores sem fio é desenvolver técnicas que otimizem recursos, principalmente energia. Geralmente, a energia é fornecida por baterias e há uma crescente escassez de recursos naturais. Além disso, é crescente a preocupação com meio ambiente e desenvolvimento sustentável. Isso tem motivado o surgimento de pesquisas para aumentar a vida útil das baterias e, consequentemente, a vida útil das RSSF.

Uma forma de realizar economia de energia é através do controle de topologia. O controle de topologia busca determinar a potência de transmissão de cada nó sensor para formar *links* de comunicação. Um *link* de comunicação existe quando dois nós estão na faixa de transmissão de energia um do outro, como visto na Figura 1. As zonas de potência de transmissão estão representadas pelos círculos. Quando um círculo de um vértice alcança outro vértice, há *link* de comunicação entre esses nós.

Ao invés de cada nó origem transmitir com a potência necessária para alcançar o destino diretamente, os algoritmos de controle de topologia tentam encontrar um subconjunto de *links* de comunicação que formem um caminho com múltiplos nós que permita a comunicação entre origem e destino e cuja potência total seja inferior à potência da transmissão direta.

A Figura 4 mostra um exemplo em que, para o nó D se comunicar diretamente com o nó E, seria necessário uma potência de valor 5, pois o custo da aresta entre eles é 5, mas pode ocorrer comunicação entre os nós percorrendo um caminho com múltiplos nós. Neste caso, o caminho D, C e E. Assim, a potência necessária para essa comunicação é de apenas 4.

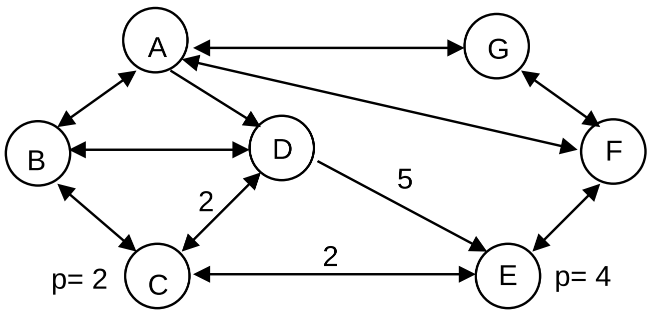


Figura 4 – Exemplo de comunicação direta e com múltiplos nós considerando valores de potência

Como consequência da redução da potência de transmissão de cada nó, os algoritmos de controle de topologia também contribuem para a redução da interferência. Um pacote de dados transmitido de um nó sensor para outro, geralmente é recebido por outros nós sensores que estão próximos do nó receptor. Isto gera interferência, que pode resultar em colisões de dados e, em seguida, retransmissões. Contudo, a redução de potência leva a redução de *links* de comunicação entre os nós e, consequentemente, falhas nos *links* de comunicação gerarão perda de conexidade. Sendo assim, os algoritmos de controle de topologia também devem se preocupar em preservar a conexidade.

Diversos trabalhos sobre controle de topologia baseado em grafos foram propostos na literatura. Foi provado que encontrar a menor potência de transmissão para cada nó, preservando a conexidade, é um problema NP-Completo (CHENG et al., 2003). Desta forma, o método de solução proposto neste trabalho utiliza-se de (meta)heurísticas para alcançar a solução de instâncias de tamanhos realísticos (até 800 nós sensores).

1.4 Objetivos

O problema a ser tratado consiste em, dado um conjunto de nós sensores com posições conhecidas, distribuídos em um espaço bidimensional, encontrar a atribuição de potência para cada nó, tal que a rede resultante possua topologia conexa e transmissões livres de interferência. Para atender as motivações mencionadas acima, o objetivo geral desta pesquisa é propor algoritmos de controle de topologia para reduzir o consumo de energia durante a comunicação entre os nós sensores. Neste trabalho, propõe-se as metaheurísticas GRASP e VND para determinar a potência de transmissão de cada nó da rede e também as metaheurísticas GRASP, VNS e GRASP-VNS, tal que a topologia resultante seja k-vértice-conexa, minimizando interferência e consumo de energia.

Os objetivos específicos deste trabalho são o de propor as metaheurísticas relacionadas acima para:

- Minimizar a potência total dos nós da rede, reduzindo o consumo de energia diretamente; minimizar a interferência dos nós da rede, reduzindo as chances de interferência nas transmissões de dados e reduzir potência e interferência simultaneamente, para os problemas FI/SMEMIT, MPT, MIMAX, MITOT, MIMAXP e MITOTP.

- Obter os mesmos resultados do modelo matemático para pequenas instâncias ($|V|$ de 10 a 50 nós) resolvidos com CPLEX (MORAES et al.,);

- Obter resultados com as metaheurísticas propostas que não puderam ser alcançados com o CPLEX.

1.5 Trabalhos relacionados

Como mostrado na Seção 1.4, neste trabalho propõe-se metaheurísticas para determinar a potência de transmissão de cada nó da rede, tal que a topologia resultante seja 2-vértice-conexa. São considerados para solução:

- Minimizar a potência total dos nós da rede, reduzindo o consumo de energia diretamente;

- minimizar a interferência dos nós da rede, reduzindo as chances de interferência nas transmissões de dados;

- Minimizar potência e interferência, simultaneamente;
- Comparação dos resultados das metaheurísticas com resultados ótimos.

1.5.1 Minimização da potência total dos nós em uma RSSF

No trabalho de Loureiro et al. [2003] são apresentados dois modelos dinâmicos de programação linear inteira mista para resolver o problema de cobertura e conexidade em RSSFs. O objetivo de Loureiro et al. [2003] é garantir a cobertura e conexidade da rede em intervalos de tempo pré-definidos, minimizando o consumo de energia.

Em Correia et al. [2005] são propostos dois modelos de ajuste de potência de transmissão em protocolos MAC para RSSF. O primeiro modelo utiliza um ajuste dinâmico para troca de informações entre os nós e o segundo modelo calcula a menor potência de transmissão em função da atenuação de sinal.

No trabalho de Silva, Gomes e Frigieri [2016] é apresentado um modelo baseado no algoritmo de Prim para calcular a mínima potência de transmissão para cada nó de uma RSSF, mantendo todos os nós conectados. Diferentes cenários são avaliados considerando o tempo computacional e a redução da potência total. Eles comparam ainda o desempenho do algoritmo para diferentes áreas e quantidades de nós na rede.

Du et al. [2019] projetam um modelo que equilibra a potência de transmissão, energia residual e conexidade da rede e investiga um modelo de controle de topologia baseado na teoria dos jogos não cooperativos. Eles propõem um algoritmo baseado na teoria dos jogos não cooperativos para controle de topologia com eficiência energética e tolerante à falhas, além de dois subalgoritmos em que o primeiro garante a conexidade da rede e o segundo, garante a biconexidade. Os resultados obtidos foram que os algoritmos podem efetivamente reduzir a potência de transmissão e prolongar a vida útil da rede e EFTCG-2 (algoritmo que mantém a biconexidade) tem capacidade de tolerância a falhas muito boa e melhora a robustez e a confiabilidade da rede.

1.5.2 Minimização da interferência dos nós em uma RSSF

Em seu trabalho, Burkhart et al. [2004] levantam uma questão: "O controle de topologia reduz interferência?". Eles mostram que os métodos tradicionais de controle de topologia com redução de interferência implícita podem falhar em alcançar efetivamente a minimização de interferência. O artigo de Burkhart et al. [2004] mostra que somente construir topologias esparsas ou somente reduzir a quantidade de conexões entre os nós em uma rede de sensores não ajuda na redução da interferência. Burkhart et al. [2004] chega à conclusão de que mensagens transmitidas entre dois nós podem afetar nós que não são vizinhos diretos. Ele propôs diversos métodos para redução da interferência em topologias conexas.

Em Moaveni-Nejad e Li [2005] é apresentado o algoritmo de árvore geradora mínima para redução da interferência máxima utilizando o modelo RIM. É provado, também, que a interferência total a partir da árvore geradora mínima é, no máximo, duas vezes os valores encontrados como ótimo.

Em Panda e Shetty [2012] é proposta uma heurística, chamada *Strong Minimum Energy Minimum Interference Topology* (SMEMIT), para minimizar o consumo total de energia através da minimização de interferência, mantendo a restrição de que a topologia resultante seja fortemente conexa. Eles provam que a potência de transmissão resultante é, no máximo, duas vezes a ideal.

Agrawal e Das [2013] propõem algoritmos para minimizar tanto a interferência máxima quanto a interferência total de uma rede de sensores sem fio. Para minimizar a interferência total, eles propõem um algoritmo ótimo com tempo de execução O(n). Eles também propõem uma heurística para minimizar a interferência total no caso de rede fortemente conectada.

Em seu trabalho, Brise et al. [2014] mostram que o objetivo é escolher potência de transmissão que minimizem interferência máxima da rede, mantendo uma comunicação assimétrica fortemente conexa. No caso unidimensional, provam que existem soluções ótimas com propriedades estruturais triviais e que essas propriedades podem ser exploradas para obter um algoritmo exato que roda em tempo quase polinomial.

Shetty e Lakshmi [2016] trabalham com o problema de minimização de interferência do receptor em uma rede de sensores sem fio. O objetivo do trabalho deles é minimizar a interferência do receptor. Eles propõem dois algoritmos, MinMax-RIP (*Minimizing Interference in Sensor networks*) e uma versão modificada do mesmo, que minimizam a interferência máxima do receptor em uma RSSF. Eles avaliam o desempenho dos algoritmos por meio de simulação. Eles modelam a RSSF como um grafo completo não direcionado.

No trabalho de Mohanty e Udgata [2020] é proposto um método para minimizar a interferência dos nós de uma rede de sensores sem fio usando diferentes esquemas de codificação e algoritmo genético. Eles mostram que o algoritmo proposto supera outros algoritmos disponíveis na literatura em termos de minimizar a máxima interferência da rede, como MI-S (*Minimizing Interference in Sensor networks*)(SHARMA et al., 2009), MinMax-RIP (*Minimizing Maximum Receiver Interference Problem*)(SHETTY; LAKSHMI, 2016), e MST (*Minimum Spanning Tree*)(CORMEN et al., 2009).

Halldórsson e Wattenhofer [2019] mostram que existem vários modelos clássicos para representar transmissões e interferências em redes sem fio, sendo que em um deles a rede sem fio é modelada por grafo. Também apresentam um modelo em que o custo de transmissões entre os nós é assimétrico, ou seja, o custo para transmissão do nó u para o nó v é diferente do nó v para o nó u. Halldórsson e Wattenhofer [2019] afirmam que, para evitar interferência, pode-se programar cuidadosamente as transmissões da rede sem fio para que as transmissões simultâneas sejam separadas no espaço ou no tempo. Também afirmam que se pode controlar a potência de transmissão, a fim de reduzir interferência. Eles mostram que aumentar a potência de transmissão de um nó remetente provavelmente aumentará a capacidade de seus pacotes de dados serem recebidos, mas também aumenta a interferência, caso haja outras transmissões simultâneas.

Abu-Affash, Carmi e Katz [2020] mostram que ter uma rede de sensores sem fio conexa com o mínimo de interferência é fundamental. Para isso, eles usam o modelo mais comum na literatura, centrado no receptor (modelo RIM). Abu-Affash, Carmi e Katz [2020] estudam o problema de atribuir um alcance de transmissão para cada sensor, de modo que o resultado seja uma rede fortemente conexa e a interferência total da rede seja minimizada. Para o caso unidimensional, mostram como resolver o problema de forma ótima em tempo $O(n^3)$. Para o caso bidimensional, mostram que o problema é NP-completo.

1.5.3 Minimização da potência e interferência dos nós em uma RSSF

Em Zhang, Zhu e Lua [2011] é proposto um algoritmo de controle de topologia, baseado no algoritmo de Árvore Geradora Mínima (*Minimun Spanning Tree*), em que minimizam a interferência e a potência simultaneamente, preservando a conexidade da rede. É um dos poucos trabalhos encontrados na literatura que buscam minimizar ambos. Eles apresentaram um algoritmo de controle de topologia denominado *Interference and Energy Aware (IEA)*. A topologia formada pelo IEA são comparadas em termos de número de *links*, interferência média e interferência máxima. Todos os grafos possuem número máximo de 100 nós.

1.5.4 Comparação de resultados de metaheurísticas com resultados ótimos

Em Nakamura et al. [2004] é proposto um modelo de Programação Linear Inteira Mista (PLIM) para realizar a organização da RSSF, de forma que sejam ativados o menor número possível de nós sensores da rede, mantendo-se a conexidade, mas estendendo-se o tempo de vida útil da rede. Como é necessário um grande esforço computacional para a resolução com a PLIM, criaram um algoritmo híbrido. Esse algoritmo é baseado no algoritmo de Dijkstra e também em uma busca (Algoritmo Genético).

Esses trabalhos da literatura nos inspiraram para desenvolver as metaheurísticas para redução de potência e minimização de interferência máxima e total em redes de sensores sem fio. Apesar deste trabalho ter se baseado nos trabalhos apresentados acima, os trabalhos (AGRAWAL; DAS, 2013), (ZHANG; ZHU; LUA, 2011), (MOAVENI-NEJAD; LI, 2005) e (HALLDÓRSSON; WATTENHOFER, 2019) foram os que mais influenciaram, pois apresentam algumas semelhanças com o trabalho proposto. O trabalho de (ZHANG; ZHU;

LUA, 2011) se assemelha com este, pois minimiza potência e interferência simultaneamente em uma RSSF. O trabalho de Halldórsson e Wattenhofer [2019] apresenta um modelo de custos assimétricos entre os nós da rede. No trabalho de Agrawal e Das [2013], é minimizado tanto a interferência máxima quanto a interferência total em uma RSSF e os autores também propõem uma heurística para isso. O trabalho de Moaveni-Nejad e Li [2005] utiliza o modelo RIM para redução de interferência máxima dos nós de uma RSSF. A principal contribuição deste trabalho é utilizar metaheurística GRASP-VNS para minimizar interferência, potência e interferência e potência, simultaneamente, além de obter resultados com as metaheurísticas propostas que não puderam ser alcançados com o CPLEX.

1.6 Contribuições

Este trabalho visa encontrar a atribuição de potência em redes de sensores sem fio propondo algoritmos que se diferenciam dos demais trabalhos, pois são utilizadas metaheurísticas para a minimização de potência, a minimização de interferência e a combinação destes.

Na literatura, as principais técnicas utilizadas para solucionar problemas de otimização de potência são os algoritmos de aproximação (ALTHAUS et al., 2006; CALINESCU et al., 2003) e algoritmos exatos (MONTEMANNI; GAMBARDELLA, 2005; DAS et al., 2004; ALTHAUS et al., 2006). Para problemas reais e de grande porte, os algoritmos exatos nem sempre encontram solução em um tempo aceitável, então são propostas heurísticas para resolução do problema.

Há trabalhos na literatura que também utilizam controle de topologia para minimizar tanto a interferência quanto o consumo de energia, como em (ZHANG; ZHU; LUA, 2011) e (PANDA; SHETTY, 2012). Esses são um dos poucos trabalhos que buscam minimizar ambos os problemas. Porém, a partir de investigação científica, foi constatado que nenhum deles utiliza a combinação da metaheurística híbrida GRASP-VNS.

Além disso, a pesquisa realizada permitiu a publicação de um artigo em congresso nacional: Yngrith Soares da Silva; Helder R. de Oliveira Rocha; Jair Adriano L. Silva; Renato Elias Nunes de Moraes; Alexandre de Souza Serrano. **Minimização de interferência com redução de potência em redes de sensores sem fio.** Em Simpósio Brasileiro de Pesquisa Operacional (SBPO), Limeira-SP, 2019 (SILVA et al., 2019). Também há um artigo em produção (MORAES et al.,).

1.7 Estrutura do trabalho

O trabalho está estruturado da seguinte forma:

- No Capítulo 2, tem-se a fundamentação teórica, que aborda diversos conceitos para o melhor entendimento desta Dissertação, como redes de sensores sem fio, teoria dos grafos, controle de topologia, otimização;
- No Capítulo 3, são mostradas a definição do problema e a metodologia é descrita, na qual são mostradas as heurísticas propostas para solução do problema quando $k = 1$;
- No Capítulo 4, são mostradas a formulação matemática e o modelo de programação inteira mista;
- No Capítulo 5, a metodologia é descrita, na qual são mostradas as heurísticas propostas para solução do problema quando $k = 2$;
- No Capítulo 6, são discutidos os resultados das metaheurísticas, além de compará-los entre si e com os resultados da formulação exata resolvida pelo CPLEX;
- Por fim, no Capítulo 7, tem-se as conclusões e os trabalhos futuros sugeridos.

2 Fundamentação teórica

Neste capítulo aborda-se alguns conceitos básicos necessários a compreensão desta Dissertação. Os conceitos apresentados estão organizados da seguinte forma: na Seção 2.1 é apresentado um resumo sobre redes de sensores sem fio; na Seção 2.2 são abordados alguns conceitos básicos de teoria dos grafos; o controle de Topologia é discutido na seção 2.3 e na seção 2.4 são discutidos alguns conceitos relacionados à otimização.

2.1 Redes de sensores sem fio

Redes de sensores sem fio (RSSF) são formadas por centenas ou milhares de nós sensores distribuídos em uma área determinada e interconectados por uma tecnologia de comunicação sem fio. Tem como principal objetivo a realização de tarefas de sensoriamento, ou seja, as RSSFs são um sistema de aquisição e manipulação de dados do ambiente.

Cada nó que compõe uma rede de sensores é composto de um subsistema computacional (memória e processador), um subsistema de sensoriamento (sensor), um subsistema de comunicação (rádio) e um subsistema de energia (bateria), como mostra a Figura 5 (SAUSEN et al., 2008).

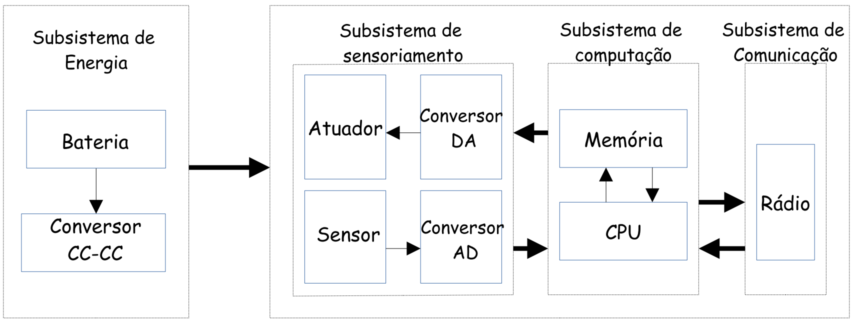


Figura 5 – Estrutura de um nó sensor

Na memória, são armazenados os dados e códigos que são executados no processador. Os sensores são a interface com o mundo real. Os sensores monitoram os parâmetros físicos do ambiente. Os sinais produzidos pelos sensores são sinais analógicos, que são convertidos em sinais digitais pelos conversores AD e são enviados ao processador (CPU). O subsistema de comunicação realiza a comunicação sem fio e pode ser formado por um dispositivo ótico ativo, passivo ou um dispositivo de rádio-frequência (RF). Quando o sistema é baseado em RF, é necessário uma antena.

O subsistema de energia é composto pela bateria e por um conversor corrente contínua-corrente contínua (CC-CC). É um subsistema delicado, pois, em geral, as baterias devem ter peso e dimensões reduzidas, o que dificulta a instalação de baterias de longa duração. A operação das baterias depende de algumas características, como o material de que é feita e a taxa de descarga. Além disso, há o fato de que as baterias, geralmente, são de difícil troca quando uma RSSF está instalada em ambientes hostis. De acordo com Sausen et al. [2008], o fim da bateria pode fazer com que uma rede de sensores seja totalmente paralisada.

O conversor CC-CC é um dispositivo que recebe uma tensão CC em sua entrada e produz uma tensão CC na saída. Geralmente a tensão de saída possui um nível diferente da tensão de entrada. É responsável por fornecer um nível de tensão constante para o subsistema de sensoriamento e sua eficiência também implica no tempo de duração da bateria (PINTO, 2004).

De acordo com Loureiro, Nogueira e Ruiz [2005], as redes de sensores sem fio podem ser classificadas com relação a configuração, coleta de dados e comunicação. Em relação a configuração, uma rede pode ser classificada como:

- Homogênea, quando todos os nós possuem as mesmas configurações (memória, CPU,...), ou heterogênea, quando as características computacionais são diferentes;
- Estacionária, quando os nós estão estáticos em uma área, ou móvel, quando os nós mudam sua posição;
- Irregular, quando a distribuição do nós não é uniforme, ou regular, quando a distribuição é uniforme.

Em relação a coleta dos dados, a rede é classificada como:

- Contínua, quando os nós coletam dados continuamente;
- Periódica, quando os nós coletam dados em intervalos regulares de tempo;
- Reativa, quando os nós coletam dados se for solicitado pela aplicação.

Já em relação a transmissão, pode ser classificada como:

- Simplex, quando os nós só transmitem os dados;
- Half-duplex, quando os nós transmitem e recebem dados, mas não fazem isso simultaneamente;
- Full-duplex, quando os nós transmitem e recebem dados simultaneamente.

Diversos autores classificam as RSSF como uma variante das Redes *Ad Hoc*, conhecidas também como *Mobile Ad Hoc Network* (MANET), mas existem diferenças entre elas (MORAES; RIBEIRO; DUHAMEL, 2009). As RSSFs possuem um número de nós maior do que nas redes *ad hoc*; há uma troca frequente da topologia (nem sempre ocasionada pela mobilidade dos nós); e possuem sérias restrições de energia.

2.2 Teoria dos Grafos

Teoria dos Grafos é a área da Matemática que estuda um grafo. O problema mais famoso de Teoria dos Grafos foi o problema das pontes de Konigsberg, proposto em 1736 por Euler. Atualmente, a Teoria dos Grafos pode ser utilizada para solucionar uma infinidade de problemas nas áreas de segurança, saúde, construção civil, automação industrial e engenharia.

Um grafo é uma estrutura $G = (V, E)$, sendo V um conjunto de elementos não vazios e E um conjunto de elementos discretos e que é uma função de V. Os elementos de V são conhecidos como vértices ou nós. Os elementos E representam uma relação de adjacência existente entre os elementos V e são conhecidos como arestas ou arcos. As arestas representam a ligação entre os nós e podem possuir custos, como, por exemplo, distância (SZWARCFITER, 2018). Dois vértices que participam de um *link* de comunicação são chamados de adjacentes.

Quando retira-se um vértice ou aresta de um grafo, este pode ser representado por G', pois foi obtido de G. Quando obtém-se um grafo $H = (V', E')$ em que $V' \subset V$ e $E' \subset E$, então diz-se que H é um subgrafo ou subestrutura de G. Existem diversas formas de se representar um grafo e uma das mais utilizadas é a matriz de adjacências $A(G)$. É uma matriz em que cada linha e coluna está associada a um vértice e relacionado como mostra a Equação (2.1).

$$a_{ij} = \begin{cases} 1, & \text{se existe } \textit{link} \text{ de comunicação entre o vértice i e o vértice j;} \\ 0, & \text{se não existe } \textit{link} \text{ de comunicação entre os vértices i e j.} \end{cases} \tag{2.1}$$

Então, se existe ligação entre um vértice i e um vértice j, a posição (i, j) na matriz recebe valor 1 se o grafo não for valorado, para representar a existência de ligação. Se não existir ligação, a matriz recebe valor 0. A matriz M, representada abaixo, exemplifica a equação 2.1. É a representação de um grafo direcionado com 3 nós e quantidade de arcos representada pela quantidade de números 1 na matriz.

$$M = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

Se o grafo for valorado, na posição (i, j) haverá o custo da ligação entre os vértices, como exemplificado pela matriz N.

$$N = \begin{bmatrix} 0 & 4 & 0 \\ 0 & 0 & 8 \\ 5 & 2 & 0 \end{bmatrix}$$

A matriz de adjacências não é a forma mais econômica de representar um grafo, pois ocupa n^2 posições de memória, mas seu uso torna menos complexo a representação de grafos grandes (NETTO, 2003).

Um importante conceito para RSSF modeladas por grafos é a conexidade. Conexidade em um grafo é representada pela possibilidade de estabelecer ligação de um vértice a outro do grafo pelas arestas existentes. Ou seja, é possuir, pelo menos, um caminho entre cada par de nós (LIN; CHEN, 2017). Se em um grafo existe, pelo menos, um vértice que não está unido ao restante do grafo por nenhuma aresta, pode-se afirmar que este grafo é desconexo. Na Figura 6, os grafos não-direcionados G1 a G3 não possibilitam a passagem de um vértice a outro qualquer, então são desconexos. Em G4, a passagem de um vértice a outro é sempre possível, sendo G4 minimal em relação a conexidade (NETTO, 2003).

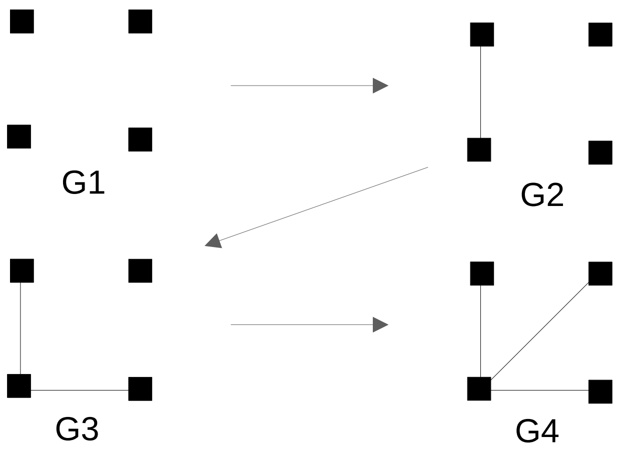


Figura 6 – Representação de conexidade em grafo não-direcionado

Se o grafo é direcionado, como representado na Figura 7, a conexidade é obtida em G5.

A conexidade permite afirmar se um grafo é conexo ou não, mas não quantifica se um grafo é mais conexo que outro. Quando um grafo é desconexo, então este pode ser decomposto em, pelo menos, dois subgrafos conexos, denominados componentes conexas. Na Figura 6, o grafo G2 está dividido em três componentes conexas.

Se a RSSF só possui um único caminho entre os pares de nós da rede e ocorrer a falha de um *link* de comunicação, essa RSSF perderá a conexidade. Então, caso seja

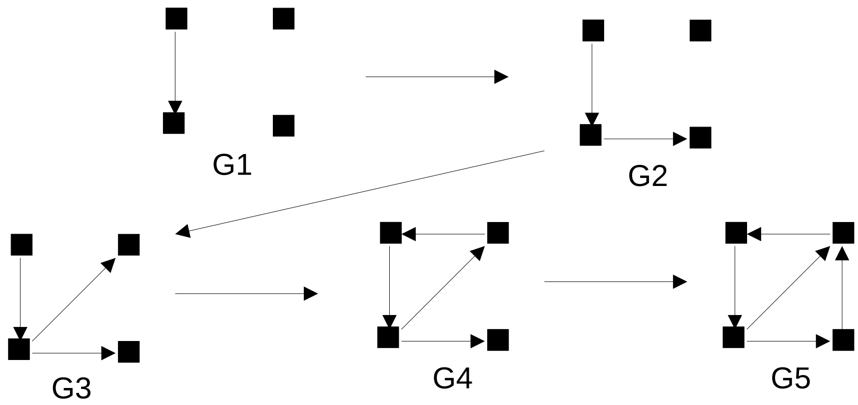


Figura 7 – Representação de conexidade em grafo direcionado

necessário uma rede tolerante a falhas, o grafo de comunicação deve possuir diversos caminhos disjuntos entre qualquer par de nós (MARINA; DAS, 2001).

Um grafo de comunicação $G = (V, E)$ é k-vértice-conexo se para dois nós quaisquer u e v, existem k caminhos que não possuem vértices em comum conectando u e v, a não ser os próprios vértices u e v, ou seja, caminhos disjuntos por vértices. Então, se mesmo após a remoção de qualquer subconjunto formado por até $k - 1$ vértices, o grafo G permanece conexo, este é considerado k-vértice-conexo. Pode-se chamar o grafo k-vértice-conexo de k-conexo, pois o grafo que é k-vértice-conexo é também k-aresta-conexo.

Na Figura 8 tem-se um exemplo de grafo que é 2-vértice-conexo, ou seja, há dois caminhos disjuntos por vértices entre todos os pares de nós. Por exemplo, para os vértices B e K, os dois caminhos estão representados pelo caminho em vermelho. Neste mesmo grafo, existem 4 caminhos arestas-disjuntos entre os vértices B e K. Então, diz-se que este grafo é 4-aresta-conexo para os vértices B e K. Os caminhos arestas-disjuntos estão representados pelos caminhos em vermelho e azul. Como afirmado acima, como o grafo é 2-vértice-conexo, pode ser chamado de 2-conexo, pois também é 2-aresta-conexo. Mas o contrário não é verdadeiro.

2.3 Controle de Topologia

Controle de Topologia é uma técnica em que o objetivo é controlar a topologia do grafo $G = (V, E)$ que representa uma RSSF de forma a melhorar alguma característica da rede. O controle de Topologia tem a finalidade de preservar propriedades da rede como, por exemplo, preservar a conexidade enquanto reduz o consumo de energia (SANTI, 2005). Como foi mencionado na Seção 1, um dos maiores problemas das RSSFs é a limitação de energia. A fonte de energia vem de baterias, que são difíceis de repor. Uma forma de otimizar o uso das baterias é reduzir a potência de transmissão de cada nó sensor, ao invés de cada nó transmitir com a potência máxima.

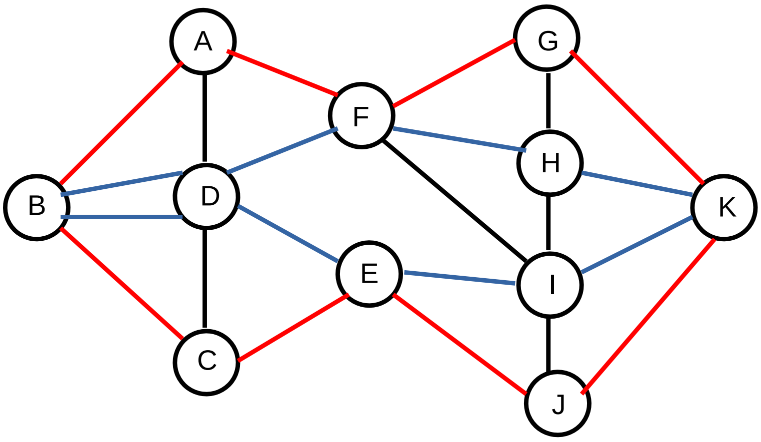


Figura 8 – Exemplo de grafo 2-conexo

O problema de controle de Topologia nas RSSFs é dividido em cobertura e conexidade (LI; YANG, 2006). Nos protocolos de Cobertura, o controle de Topologia busca reduzir o consumo de energia e preocupa-se em como os nós sensores são distribuídos em uma área de forma a aumentar o sensoriamento da área considerada. Já os protocolos de Conexidade se preocupam em realizar a troca confiável de dados entre os nós sensores. Assim, controle de Topologia é a técnica de realizar uma comunicação eficiente na rede, reduzindo o consumo de energia e mantendo a sua conexidade.

Se os nós sensores utilizam a potência máxima P_{max} para transmissão de dados, tem-se o grafo G_{max}. Então o controle de Topologia busca um subgrafo G' tal que (NETO,):

- Vértices de G_{max} = Vértices de G';
- Se existe um caminho entre u e v em G_{max}, também existe um caminho entre u e v em G';
- A potência dos nós em G' é menor que a potência dos nós em G_{max};
- O número de arestas em G' é menor do que o número de arestas em G_{max}.

Na Figura 9, vê-se um exemplo na qual diminui-se o número de arestas, reduzindo a potência do nó. O nó a transmite dados para os nós b e c, possuindo uma potência p_2. Considerando que a soma do custo da aresta (a, b) com o custo da aresta (b, c) é menor do que o custo (a, c), o algoritmo de controle de Topologia elimina a aresta (a, c). Assim, a potência do nó a é reduzida para p_1 mantendo a comunicação para os nós b e c, como pode ser visto na Figura 9.

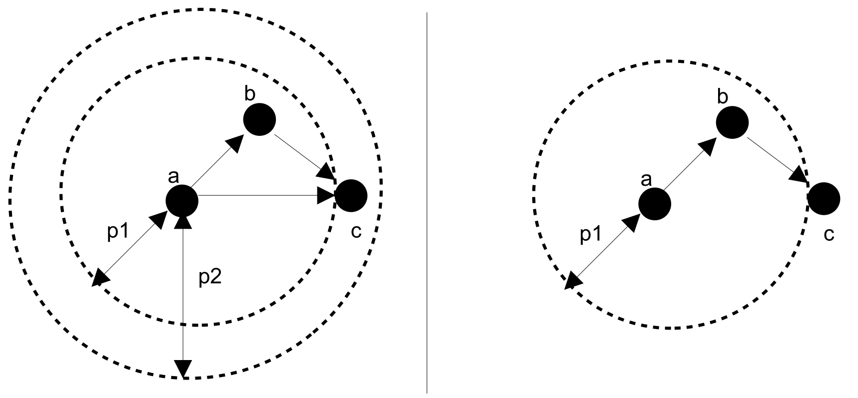


Figura 9 – Ajuste de potência pelo algoritmo de controle de Topologia

2.4 Otimização

A otimização envolve a maximização ou a minimização de uma função de uma ou mais variáveis, chamada de função objetivo. Função objetivo representa o objetivo principal daquele problema, o que deseja ser atingido. É acompanhada de um conjunto de restrições e variáveis de decisão. É representada, matematicamente, por uma função f com domínio S, chamada de função custo ou função de avaliação (LUZ; BECCENERI; VELHO, 2008).

Em um problema de minimização, dada uma função f, temos que encontrar $s* \in S$ tal que $f(s*) \leq f(s)$, para todo $s \in S$. Num problema de maximização, temos que encontrar $s* \in S$ tal que $f(s*) \geq f(s)$, para todo $s \in S$ (LUZ; BECCENERI; VELHO, 2008).

Os algoritmos utilizados para otimização podem ser determinísticos ou probabilísticos (FARIA, 2017). Algoritmos determinísticos são caracterizados por produzir a mesma solução sempre que as variáveis de entrada não forem alteradas (forem repetidas). Já os algoritmos probabilísticos produzem soluções diferentes, mesmo que as variáveis de entrada sejam repetidas em diversos testes, pois possuem eventos pseudoaleatórios, como, por exemplo, a geração de um número randômico (LUZ; BECCENERI; VELHO, 2008).

Os algoritmos de otimização atuam em um espaço de busca ou espaço de soluções, isto é, um conjunto que contém todas as soluções do problema. A etapa principal em problemas de otimização é a definição do espaço de busca. A melhor solução de uma área específica do espaço é chamada de ótimo local, enquanto que a melhor solução do problema é conhecida como ótimo global (FREITAS et al., 2014). De acordo com Melián, Pérez e Vega [2003], três características são significativas nestes algoritmos: soluções inviáveis, função objetivo e critério de parada.

A função objetivo é importante quando a avaliação da solução é computacionalmente custosa. O critério de parada define o momento em que o algoritmo será finalizado e pode ser definido de acordo com o tempo gasto ou o número de iterações sem melhoria, por exemplo (MELIÁN; PÉREZ; VEGA, 2003).

Dois processos têm grande importância nestes algoritmos: a diversificação e a intensificação. Com a diversificação, o processo de busca mantém uma aleatoriedade para abranger o espaço de busca. Com a intensificação, reforça o desempenho das boas soluções encontradas. O desafio destes algoritmos é manter o equilíbrio entre esses processos.

3 Heurísticas aplicadas à RSSF

Neste capítulo são apresentadas a definição do problema e são propostas metaheurísticas para reduzir o consumo de potência em RSSF. São explicados o algoritmo guloso sobre custo de envio, o algoritmo guloso sobre cobertura e são apresentadas as metaheurísticas FI/SMEMIT, GRASP e VND.

3.1 Definição do problema

As RSSF podem ser representadas através de um grafo $G = (V, E)$, onde V é o conjunto de nós sensores da rede e E é o conjunto de arestas que representam os *links* de transmissão entre os nós (SHETTY; LAKSHMI, 2016; HALLDÓRSSON; WATTENHOFER, 2019). O conjunto de nós, juntamente com as arestas, estabelece um grafo de comunicação.

Para cada aresta, é conhecido um custo não-negativo $c(u, v)$. Há, também, uma potência de transmissão p_u associada a cada nó u pertencente a V. De acordo com Rappaport [2001], para que um nó u transmita dados de forma correta para um nó v, é necessária uma potência:

$$p = d^{\varepsilon}_{u,v}.q_v \tag{3.1}$$

na qual d é a distância euclidiana entre o nó transmissor u e o nó receptor v, ε é o expoente de perda no caminho e q_v é o limiar de potência do receptor para detecção do sinal. O valor do expoente ε pode variar entre 2 e 4. Neste trabalho, utilizou-se $\varepsilon = 2$, pois é o valor considerado para o espaço livre. Assumindo um modelo de perda no caminho determinístico, $q_v = 1$, então:

$$p = d^{\varepsilon}_{u,v} \tag{3.2}$$

Assim, um sinal transmitido por um nó u é recebido pelo nó v se, e somente se, a potência de transmissão do nó u é, pelo menos, igual a $c(u, v)$, como mostra a equação:

$$p_u \geq c(u, v) \tag{3.3}$$

Um nó u só alcança um nó v se a potência de transmissão do nó u for suficiente para alcançar o nó v. A potência total da RSSF é obtida somando-se a potência de cada nó do grafo.

Uma maneira de analisar o problema de consumo de potência seria se os grafos tivessem entradas simétricas. Neste caso, o custo da aresta para estabelecer a transmissão entre os nós u e v é: $c(u, v) = c(v, u)$. Esta forma só é válida para ambientes em que não há nenhum obstáculo entre os receptores. Na prática, sabe-se que podem ocorrer espalhamentos, difrações, reflexões. Então, a comunicação entre os nós u e v da rede, provavelmente, será modelado por custos assimétricos. Assim, $c(u, v) \neq c(v, u)$.

Pode ocorrer de um nó ter uma potência de transmissão que faz alcançar nós que não necessariamente são os nós destino da mensagem e, consequentemente, essa transmissão causa interferência se realizada ao mesmo tempo em que outros nós estão trocando mensagens, como foi mostrado na Figura 1.

Na Figura 10, os nós u e w estão enviando mensagens entre si. Suponha que o nó v esteja recebendo uma mensagem do nó b no mesmo instante que u e w se comunicam. O nó v, por estar conectado aos nós u e w, sofrerá interferência, corrompendo a mensagem que o nó v iria receber do nó b, aumentando o consumo de energia para um novo envio da informação de b para v. Formalmente, a interferência de um nó v em uma rede representado pelo grafo $G = (V, E)$ é definido por:

$$I(v) = |\{u \in V \setminus \{v\} : (u, v) \in E\}|. \tag{3.4}$$

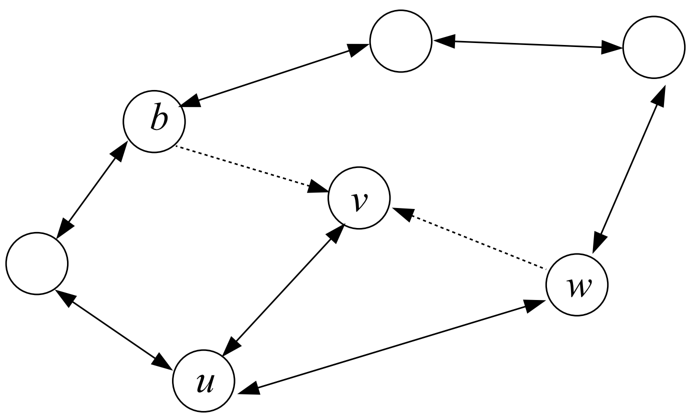


Figura 10 – Modelo de rede de sensores sem fio estacionária com 7 nós

De acordo com Bilò e Proietti [2008], os dois principais modelos de interferência existentes são o Modelo Interferência-Emissor (SIM - *Sender Interference Model*) e o Modelo Interferência-Receptor (RIM - *Receiver Interference Model*). No modelo SIM, a interferência está relacionada ao grau de saída do nó. Analogamente, no modelo RIM, a interferência em um nó está relacionada ao grau de entrada do nó.

Na Figura 10, se o modelo SIM for adotado, a interferência do nó b é 3. Mas, se o modelo RIM for adotado, a interferência neste mesmo nó será 2. A interferência máxima corresponde a maior interferência que um nó contido no grafo pode ter. Na

Figura 10, adotando o modelo RIM, a interferência máxima é 3, sendo os nós u e v com estes valores de interferência. Adotando o modelo RIM, a interferência total é a soma de todas as interferências dos nós sensores. No exemplo da Figura 10, a interferência total será: 2+2+2+2+2+3+3=16. De acordo com Yilmaz, Dagdeviren e Erciyes [2011], o modelo RIM é mais robusto do que o modelo SIM para cálculos de interferência máxima.

Como a potência de um nó u é a potência necessária para alcançar o nó mais distante com o qual ele quer comunicar-se diretamente, a potência total é obtida somando-se a potência de cada nó do grafo. Panda e Shetty [2012] demonstram que é importante encontrar o conjunto de atribuição de potências p, tal que: G seja conexo, a potência total da RSSF seja minimizada e as interferências sejam minimizadas.

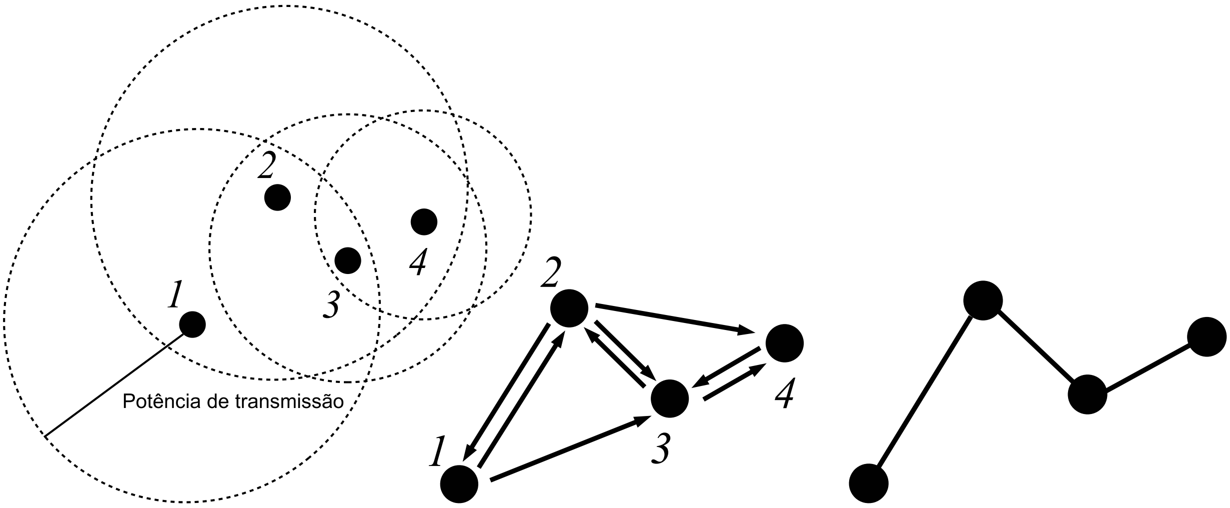
Desta forma, o problema consiste em, a partir de um grafo completo G assimétrico e do parâmetro $k \geq 1$, encontrar um subgrafo G' conexo e 2-conexo direcionado, tal que todo nó seja capaz de enviar e receber informações de todos os nós da rede, visando reduzir o consumo de energia da rede. O problema a ser resolvido por esse trabalho é encontrar um conjunto de atribuição de potências $p : V \rightarrow R+$ tal que:

- $max_{u \in V} I(u)$ é mínimo;
- $\sum_{v \in V} I(v)$ é mínimo;
- $\sum_{u \in V} p_u$ é mínimo.

onde $I(u)$ é a interferência de um nó no u modelo RIM, $I(v)$ é a interferência total e p_u é a potência do nó u.

Há duas topologias que podem ser utilizadas para representar a restrição de k-conexidade, sendo elas: Topologia unidirecional e Topologia bidirecional. Na topologia unidirecional, todos os arcos formados pela atribuição de potência p no grafo G são considerados para a restrição de k-conexidade. O uso de *links* de comunicação unidirecionais em uma RSSF é questionável (SANTI, 2005). *Links* unidirecionais geram atrasos para manipulação das ligações pelos algoritmos de roteamento. Topologias bidirecionais geram um custo adicional, mas oferecem melhor desempenho (MARINA; DAS, 2002). Em uma topologia bidirecional, uma aresta é usada para *link* de comunicação para estabelecimento da restrição de k-conexidade se um nó u está dentro do alcance de um nó v e se o nó v também está no alcance de transmissão do nó u. As Figuras 11a, 11b e 11c exemplificam uma RSSF e as topologias unidirecionais e bidirecionais da RSSF, respectivamente.

Tanto o problema de consumo mínimo de potência com topologia unidirecional como com topologia bidirecional são NP-hard (CALINESCU; WAN, 2006). Em cenários reais, os enlaces de transmissão das RSSF são assimétricos, ou seja, o custo de transmissão



(a) Exemplo de uma RSSF (b) Topologia unidirecional (c) Topologia bidirecional

Figura 11 – Exemplo de uma RSSF e as variações de topologias unidirecional e bidirecional

Fonte: (MORAES; RIBEIRO; DUHAMEL, 2009)

$c(u, v)$ pode ser diferente do custo $c(v, u)$. Moraes, Ribeiro e Duhamel [2009] apresentam uma Função Gulosa Estática para solucionar o problema de enlaces assimétricos em RSSF. Conforme a equação:

$$c(uv) = c(u, v) + c(v, u) \tag{3.5}$$

Ou seja, os custos $c(u, v)$ e $c(v, u)$ são somados obtendo-se um único valor para o enlace, tornando-se $c(uv)$. Assim, o grafo pode ser tratado como uma rede simétrica, permitindo a aplicação de diversos algoritmos gulosos, como, por exemplo, o algoritmo sobre custo de envio que será apresentado na Seção 3.2 e o algoritmo sobre cobertura que será apresentado na Seção 3.3 com adaptações para o problema estudado.

3.2 Árvore geradora mínima sobre custo de envio

A implementação é baseada no algoritmo de (PRIM, 1957) para construir a árvore geradora mínima. Foi utilizada a Função Gulosa Estática para tornar a rede simétrica. O pseudocódigo do Algoritmo 1 mostra o funcionamento do algoritmo de árvore geradora mínima sobre custo de envio.

O algoritmo recebe como entrada o grafo completo $G = (V, E)$ representando uma RSSF. Na linha 2, as potências de todos os nós são inicializadas em zero. Na linha 3, é construída uma árvore geradora mínima utilizando-se o algoritmo de Prim modificado, pois, neste algoritmo, a grandeza utilizada para comparação são os custos $c(uv)$. No laço das linhas 4 a 6, é atribuído a p_u o custo necessário para cada nó u alcançar o nó mais

Algoritmo 1: Algoritmo de construção da árvore geradora mínima sobre custo de envio

Entrada: $G = (V, E)$, onde V é o conjunto de nós e E é o conjunto de arestas;
$p_u \leftarrow 0, \forall u \in V$;
Encontrar árvore geradora mínima T comparando os custos: $c(uv) = c(u, v) + c(v, u)$;
for u *em* V **do**
  $p_u \leftarrow \max_{(v \in V\{u\})}\{c(u, v) : (u, v) \in E(p)\}$;
end for
Saída: Atribuição de potências p e $G(p) = (V, E(p))$;

distante ao qual u está ligado. Por utilizar o algoritmo de Prim como base, a complexidade deste algoritmo é $O(|E|log|V|)$.

3.3 Árvore geradora mínima sobre cobertura

Este algoritmo também é baseado no algoritmo de Prim (PRIM, 1957), mas considera a cobertura das arestas para construção da árvore geradora mínima. De acordo com Burkhart et al. [2004], cobertura de uma aresta é definida como a quantidade de nós que são perturbados quando os nós u e v se comunicam. Enlaces com menores valores de cobertura perturbarão menos nós, reduzindo a interferência, comparado a enlaces com maior cobertura. Formalmente, tem-se:

$$cob(u, v) := |\{w \in V | w \text{ é alcançado por } u\} \cup \{w \in V | w \text{ é alcançado por } v\}| \tag{3.6}$$

Entretanto, utilizar o método $cob \cup (u, v)$ pode gerar falhas, pois quando dois nós estão se comunicando e ambos interferem em um mesmo nó, apenas uma unidade será adicionada à cobertura do enlace. Então, utilizou-se o seguinte cálculo de cobertura:

$$cob(u, v) := |\{w \in V | w \text{ é alcançado por } u\} + \{w \in V | w \text{ é alcançado por } v\}| \tag{3.7}$$

O pseudocódigo do Algoritmo 2 mostra a construção da árvore geradora mínima utilizando cobertura como comparação entre arestas.

Na linha 2, o algoritmo inicializa as coberturas de cada enlace zerando-as. O laço que inicia na linha 3 e termina na linha 12 checa se o custo de envio da mensagem de u para w é menor que o custo de envio de u para v. Se sim, o nó w é alcançado pelo nó u, então a cobertura do enlace $cob(u, v)$ é incrementada em 1. De forma análoga, nas linhas 8 a 10, checa-se se o nó w é alcançado por v. Se sim, é incrementado em 1 o valor da cobertura $cob(v, u)$. Na linha 13, as potências de todos os nós são inicializadas em zero.

Algoritmo 2: Algoritmo de construção da árvore geradora mínima sobre cobertura

1 **Entrada:** $G = (V, E)$, onde V é o conjunto de nós e E é o conjunto de arestas;
2 $cob(u, v) \leftarrow 0, \forall\ (u,v) \in V$;
3 **for** (u, v) *em* E **do**
4   **for** w *em* V **do**
5   **if** $c(u, w) \leqslant c(u, v)$ **then**
6             $cob(u, v) + +$;
7   **end if**
8   **if** $c(v, w) \leqslant c(v, u)$ **then**
9             $cob(v, u) + +$;
10   **end if**
11   **end for**
12 **end for**
13 $p_u \leftarrow 0, \forall u \in V$;
14 Encontrar árvore geradora mínima T comparando os custos: $c(uv) = c(u, v) + c(v, u)$;
15 **for** u *em* V **do**
16     $p_u \leftarrow \max_{(v \in V\{u\})}\{c(u, v) : (u, v) \in E(p)\}$;
17 **end for**
18 **Saída:** Atribuição de potências p e $G(p) = (V, E(p))$ conexo;

Na linha 14, é construída uma árvore geradora mínima utilizando-se o algoritmo de Prim comparando as coberturas. No laço das linhas 15 a 17, é atribuído o valor de potência a cada nó sensor na árvore. Este algoritmo possui complexidade $O(|E|log|V|)$.

Os algoritmos apresentados acima obtêm soluções viáveis para o problema de redução de potência, mas nem sempre produzem a melhor solução para o problema. Para melhorar as soluções encontradas, usa-se a busca local.

Busca local é um procedimento de exploração sistemática de vizinhança e é empregada para melhorar o resultado de uma solução s. Dado um espaço de soluções S, a vizinhança $N(s)$ de uma solução $s \in S$ é um conjunto de soluções $N(s) \subseteq S$. Uma busca local é um procedimento que explora sistematicamente a vizinhança $N(s)$ de uma solução s. Sendo assim, uma busca local explora uma vizinhança de uma solução já conhecida s à procura de uma solução $s' \in N(s)$ melhor do que s. Quando s' é melhor que s, atualiza-se a solução corrente para s'. Quando a busca local não consegue melhorar a solução corrente, alcançou-se um ótimo local. Desta forma, não é possível melhorar a solução partindo do mesmo s e explorando a mesma $N(s)$ (HANSEN; MLADENOVIC, 2007).

Existem buscas locais que são do tipo melhor aprimorante e também há buscas locais que são do tipo primeiro aprimorante. A busca local melhor aprimorante só atualiza a solução corrente para s' após percorrer todos os vizinhos da solução corrente s e descobrir que s' é a melhor solução entre todos os vizinhos possíveis. Já a busca local do tipo primeiro aprimorante atualiza a solução corrente imediatamente após encontrar a primeira solução na vizinhança melhor do que a atual (HANSEN; MLADENOVIC, 2007). Uma estratégia de busca local do tipo primeiro aprimorante é proposta a seguir.

3.4 Greedy Randomized Adaptive Search Procedure - GRASP

A metaheurística GRASP foi proposta por Feo e Resende [1995]. É um processo composto de duas fases: construção da solução inicial e busca local. Na fase de construção, uma solução inicial viável é formada utilizando o algoritmo a ser apresentado na subseção 3.4.1. Na fase de busca local, procura-se melhorar a qualidade da solução inicial. A busca local será apresentada na subseção 3.4.2.

Para ilustrar a implementação da heurística GRASP apresenta-se o Algoritmo 3 que mostra o pseudocódigo do GRASP.

Algoritmo 3: Algoritmo GRASP

1 **Entrada:** Grafo $G = (V, E)$, parâmetro α, critério de parada;
2 $f(s*) \leftarrow \infty$
3 **for** *critério de parada não satisfeito* **do**
4     $s \leftarrow$ `Algoritmo_Guloso_Randomizado`(G,α)
5     $s'' \leftarrow$ `FI/SMEMIT`(s)
6   **if** *f(s") < f(s*)* **then**
7         $f(s*) \leftarrow f(s'')$
8   **end if**
9 **end for**
10 **Saída:** $f(s*)$

Na linha 2, o melhor valor da função objetivo é inicializado com infinito, porque ainda não há solução inicial criada e como o objetivo é reduzir potência, é necessário usar um valor bem alto para ir reduzindo. O laço das linhas 3 a 9 é repetido até que o critério de parada seja alcançado. Na linha 4, uma solução é construída através do Algoritmo 4. Na linha 5, executa-se o Algoritmo 5. Na linha 7, testa-se se houve melhoria na solução. Se sim, então a solução corrente é substituída pela melhor solução, já que a busca local usada é do tipo primeiro aprimorante.

3.4.1 Randomized Prim

Os algoritmos gulosos randomizados são um tipo de algoritmo guloso. Da mesma forma que um algoritmo puramente guloso, na execução do Algoritmo Guloso Randomizado, as soluções são construídas de forma progressiva desde o início do algoritmo, iteração a iteração. Em cada iteração, um novo elemento é adicionado à solução em construção até que uma solução viável completa seja formada (MORAES; RIBEIRO; DUHAMEL, 2009). Diferentemente do algoritmo puramente guloso, que é determinístico e sempre escolhe o melhor elemento de acordo com a função gulosa utilizada, um Algoritmo Guloso Randomizado seleciona um elemento aleatoriamente de uma lista contendo elementos candidatos que mantém a viabilidade da solução. Dessa forma, nem sempre o melhor elemento será selecionado. O objetivo dessa estratégia é diversificar a solução construída.

O algoritmo receberá como entrada um grafo $G(V, E)$ com entradas assimétricas e a função gulosa, explicada na Seção 3.1, realiza o cálculo previsto. A partir dessa entrada, são atribuídas potências de transmissão para cada nó do grafo. O algoritmo guloso randomizado seleciona arestas aleatoriamente em uma Lista Restrita de Candidatos (LRC). A cada iteração do algoritmo guloso randomizado, ordena-se em ordem crescente o conjunto de arestas candidatas a entrarem na solução e forma-se uma Lista de Candidatos (LC). A partir de um valor pré-definido α, a LC é restringida, obtendo-se a LRC. A LRC é formada pelas arestas $(u, v) \in$ LC, que estão dentro do valor limite estabelecido por:

$$lim = c_{min} + \alpha(c_{max} - c_{min}), \tag{3.8}$$

onde $\alpha \in [0.0, 1.0]$, c_{min} é o custo da menor aresta que faz parte de LC e c_{max} é o custo da maior aresta que faz parte de LC, obtendo-se a LRC. Se $\alpha = 0.0$, o valor limite será igual a c_{min} e a aresta escolhida será a de menor custo, fazendo o algoritmo guloso randomizado ser igual a um algoritmo puramente guloso. Em um algoritmo puramente guloso, a aresta de menor custo $c(u, v)$ é a que seria selecionada para fazer parte da solução.

Se $\alpha = 1.0$, a escolha dos candidatos para compor a LRC será estritamente aleatória e será escolhida qualquer aresta que não esteja na solução. A LRC deve ser formada por arestas (u, v) válidas. Neste algoritmo, uma aresta (u, v) é considerada válida quando conecta um nó u pertencente à solução a um nó v que está fora da solução. Então, através da formação da LRC, o algoritmo guloso randomizado consegue produzir diferentes soluções a cada execução.

Para $k = 1$, o valor de α é 0,3 e foi escolhido empiricamente. Assim, somente as arestas com custos menores que 30% da diferença entre o menor e maior custo serão adicionadas a LRC. O algoritmo seleciona uma aresta aleatoriamente na LRC para compor a solução. Este algoritmo possui complexidade $O(|E|log|V|)$. No Algoritmo 4 é apresentado o pseudocódigo do Randomized Prim.

Na linha 1, recebe-se como entrada um grafo $G(V, E)$ e o parâmetro α. Ou seja, é recebido o conjunto de nós V e a distância entre os nós. Na linha 2, as potências de todos os nós do grafo G são inicializadas em zero.

Na linha 3, o conjunto de arestas que não fazem parte da solução corrente E_c é inicializado com todas as arestas E do grafo G e o conjunto de arestas que fazem parte da solução corrente E_f é inicializado vazio na linha 4. Na linha 5, o conjunto de vértices V', que farão parte da solução, é inicializado em zero. Na linha 6, o primeiro nó é escolhido e inserido no conjunto V'. O laço das linhas 7 a linha 20 é executado até que todos os nós

Algoritmo 4: Randomized Prim

1 **Entrada:** $G = (V, E)$, onde V é o conjunto de nós e E é o conjunto de arestas e parâmetro α;
2 $p_u \leftarrow 0, \forall u \in V$;
3 $E_c \leftarrow E$;
4 $E_f \leftarrow 0$;
5 $V' \leftarrow 0$;
6 $V' \leftarrow V' \cup \{u\}$;
7 **while** $V' \neq V$ **do**
8     $LC \leftarrow$ ordenação de E_c por custo;
9     $c_{min} \leftarrow$ custo da menor aresta de LC;
10     $c_{max} \leftarrow$ custo da maior aresta de LC;
11     $lim = c_{min} + \alpha(c_{max} - c_{min})$;
12     $LRC \leftarrow (u, v), \forall(u, v) \in LC$ tal que $c(u, v) \leq lim, u \in V', v \notin V'$;
13   seleciona-se uma aresta aleatória (u, v) de LRC;
14   **if** (u, v) *é aresta válida* **then**
15   remover (u, v) de E_c;
16   adicionar (u, v) a E_f;
17         $V' \leftarrow \{v\}$;
18   atualizar p_u e p_v;
19   **end if**
20 **end while**
21 **Saída:** Atribuição de potências p e $G(p) = (V, E(p))$ conexo;

façam parte da solução em construção.

A cada iteração do laço da linha 7 a 20, primeiro, na linha 8 ordena-se as arestas E_c que não estão na solução e esta lista é então atribuída a LC. Nas linhas 9 e 10, é calculado o menor e maior custo das arestas em LC. Na linha 11, calcula-se lim. Na linha 12, atribui-se a LRC todas as arestas de LC que estão abaixo de lim e que sejam arestas válidas.

Na linha 14, a aresta (u, v) pode ser incorporada à solução corrente. Assim, na linha 15, remove-se (u, v) de E_c e nas linhas 16 e 17, a solução corrente é atualizada pela adição da aresta (u, v) selecionada a E_f e pela adição do nó $v \in (u, v)$ a V'. Na linha 18, as potências dos nós u e v são atualizadas.

3.4.2 *Fast Improved Strong Minimum Energy Minimum Interference Topology (FI/SMEMIT)*

O algoritmo Guloso Randomizado obtém soluções viáveis, mas não são necessariamente as soluções ótimas. Então, utiliza-se algoritmos de busca local para melhorar as soluções encontradas pelo guloso randomizado e aumentar as chances de se encontrar a solução ótima.

É uma adaptação do algoritmo sugerido por (PANDA; SHETTY, 2012). A partir da solução inicial, remove-se uma aresta, de cada vez, da solução corrente e insere-se outra aresta que não está na solução. A troca de k arestas simultaneamente é chamado de k-movimento, então este algoritmo é 1-movimento. Na retirada de uma aresta, o grafo é dividido em duas componentes conexas. Para eliminar o teste de conexidade e reduzir o tempo computacional gasto, os nós que serão conectados pela nova aresta devem estar cada um em uma componente conexa diferente.

Este método manuseia, na entrada, os resultados obtidos pelo Algoritmo 4. Ordena-se as arestas da solução obtida por ordem crescente de cobertura. A partir da solução inicial, 1-movimentos serão efetuados, retirando a aresta com maior cobertura e acrescentando uma outra aresta. O objetivo deste algoritmo é obter melhorias na: interferência máxima; interferência total sem que haja prejuízo na interferência máxima; potência total sem que haja prejuízo nas interferências máxima e total. O Algoritmo 5 detalha o FI/SMEMIT.

Algoritmo 5: Algoritmo FI/SMEMIT

1 **Entrada:** $G = (V, E)$, onde V é o conjunto de nós e E é o conjunto de arestas e parâmetro α;
2 $p_u \leftarrow 0, \forall u \in V$;
3 $p_a \leftarrow p$;
4 $p_* \leftarrow p$;
5 $E_{ord} \leftarrow Ordenar E' por cobertura$;
6 $melhoria \leftarrow verdadeiro$;
7 **while** $melhoria = verdadeiro$ **do**
8     $V_1, V_2 \leftarrow$ remove (u, v) de E_{pa};
9   **for** x *em* V_1 **do**
10   **for** y *em* V_2 **do**
11             $G^+(V, E(p^+) \leftarrow$ Conecta x $\in \mathrm{V}_1$ e y $\in \mathrm{V}_2$;
12   **if** $G(p^+)$ *é melhor que* $G(p^*)$ **then**
13                 $p_* \leftarrow p^+$;
14                 $melhoria \leftarrow verdadeiro$;
15   ir para linha 19;
16   **end if**
17   **end for**
18   **end for**
19     $p_a \leftarrow p^*$;
20 **end while**
21 **Saída:** Atribuição de potências p e $G(p) = (V, E(p))$

3.5 Variable Neighborhood Descent (VND)

É uma metaheurística proposta por (MLADENOVIC, 1995). O algoritmo inicializa com uma configuração inicial de boa qualidade. Neste caso, foi utilizado o Algoritmo

Randomized Prim para gerar essa configuração inicial. Após, realiza-se uma busca local, ou seja, são gerados todos os vizinhos da solução dentro da sua estrutura de vizinhança. No algoritmo gerado, os vizinhos são criados retirando arestas e acrescentando novas arestas na solução.

Na primeira estrutura de vizinhança, é realizado 1-movimento. Cada um dos vizinhos é comparado com a solução atual. Se a solução fornecida pela estrutura vizinha é melhor, então esta será a nova solução atual, a busca é reiniciada e k se torna igual a 1. Quando não se encontra uma solução melhor do que a atual, então passa-se para a próxima estrutura de vizinhança. A ordem de transição das estruturas é sistemática, então a próxima estrutura de vizinhança é k igual a 2. O processo de busca se inicia retirando-se k arestas e inserindo-se k novas arestas no grafo da rede através da função *Shake*. A busca local é a apresentada no Algoritmo 5. O objetivo deste algoritmo também é obter melhorias na: (i) interferência máxima, (ii) interferência total sem que haja prejuízo na interferência máxima, (iii) potência total sem que haja prejuízo nas interferências máxima e total. O pseudocódigo do VND está mostrado no Algoritmo 6.

Algoritmo 6: VND básico

```
1  Entrada: s (solução inicial), k_max (número de estruturas de vizinhança)
2  Saída: s (solução)
3  k ← 1
4  for k ≤ k_max do
5  |   s' ← shake(s, k)
6  |   s'' ← BuscaLocal(s')
7  |   if f(s”) < f(s) then
8  |   |   s ← s''
9  |   |   k ← 1
10 |   end if
11 |   k ← k + 1
12 end for
```

Na linha 3, a variável k é inicializada com 1. O algoritmo inicia recebendo uma solução inicial s e o número de estruturas de vizinhanças k_{max}. No laço das linhas 3 a 10, a função *Shake* (linha 6) realiza movimentos aleatórios na solução s dentro da vizinhança k e gera uma nova solução s'. Na linha 7, a função *BuscaLocal* recebe como entrada a solução s' e gera uma nova solução s''. A solução s'' é uma solução melhor do que a solução s', obtida no Algoritmo 11.

4 Modelos de Programação Inteira Mista

Neste capítulo é apresentada a formulação matemática para solucionar o problema de consumo de energia através da programação linear inteira mista baseado em fluxo de multicomodidades.

São propostas as metaheurísticas GRASP, VNS e GRASP-VNS para determinar a potência de transmissão de cada nó da rede, tal que a topologia resultante 2-vértice-conexa (tolerante a falhas), minimizando interferência e consumo de energia (MORAES et al.,). Então, os problemas de minimização discutidos serão:

- Potência Total Mínima (MPT), onde a função objetivo é a mínima potência total da rede;
- Interferência Máxima Mínima (MIMAX), onde a função objetivo é a mínima interferência máxima experimentada por qualquer nó da rede;
- Interferência Total Mínima (MITOT), onde a função objetivo é a mínima interferência total experimentada pelos nós da rede;
- Interferência Máxima Mínima com mínima Potência total (MIMAXP), onde a função objetivo é a mínima interferência máxima experimentada por qualquer nó da rede, além da mínima potência total da rede;
- Interferência Total Mínima com mínima Potência Total (MITOTP), onde a função objetivo é a mínima interferência total experimentada pelos nós da rede, além da mínima potência total da rede.

4.1 Formulação matemática

A resolução exata do problema de consumo mínimo de potência com restrição de k-conexidade baseado em fluxo de comodidades é apresentada para solucionar o problema com topologia bidirecional e entradas inicialmente assimétricas (MORAES; RIBEIRO; DUHAMEL, 2009).

Uma forma eficiente de formular as restrições de k-conectividade consiste em definir um conjunto C de $\lceil k|V|/2 \rceil$ comodidades com demanda de uma unidade (MORAES; RIBEIRO; DUHAMEL, 2009). Para cada comodidade $c \in C$, representa-se por $o(c)$ a origem e por $d(c)$ o destino. Para qualquer nó $i \in V$ e qualquer comodidade $c \in C$, seja $D_c(i) = -k$ se $i = o(c)$, $D_c(i) = +k$ se $i = d(c)$ e $D_c(i) = 0$ caso contrário.

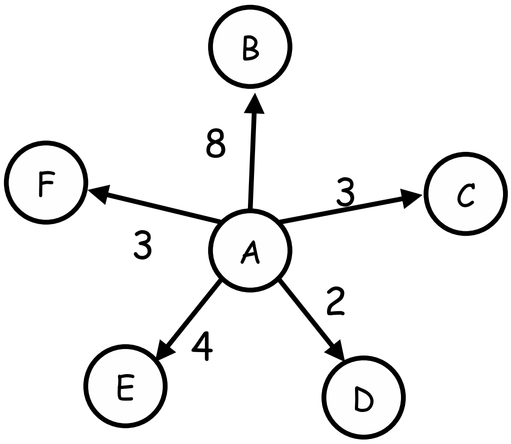


Figura 12 – $Q_A = [2, 1, 1, 4]$, $T_A^1 = [D]$, $T_A^2 = [C, F]$, $T_A^3 = [E]$ e $T_A^4 = [B]$

Adaptado de: (MORAES, 2009)

A variável discreta f_{ij}^c representa o fluxo de comodidade c através do arco (i, j). A variável binária f_{ij}^c é igual a um se o arco (i, j) é usado pela comodidade c para comunicação do nó i para o nó j e zero, caso contrário.

Seja $Q_i = [q_i^1, \ldots, q_i^{\phi(i)}]$ uma lista finita de elementos, que são incrementos sucessivos e cumulativos de níveis de potência atribuídos ao nó i, para cada $i \in V$. Além disso, $\phi(i) \leq |V| - 1$ e para qualquer $\ell = 1, \ldots, \phi(i)$, define-se T_i^ℓ como o conjunto de novos nós alcançáveis do nó i se um incremento adicional q_i^ℓ é adicionado à sua potência atual. A Figura 12 exemplifica os conjuntos Q_i e T_i^ℓ.

A variável binária x_i^ℓ tem valor um se houver um nó $j \in T_i^\ell$ tal que um arco direcional (i, j) é usado para comunicaação de i a j, caso contrário recebe valor zero. Também define-se $\bar{\ell}(i) \in \{1, \ldots, \phi(i)\}$ de forma que $\sum_{\ell=1}^{\bar{\ell}(i)-1} |T_i^\ell| < k$ e $\sum_{\ell=1}^{\bar{\ell}(i)} |T_i^\ell| \geq k$. Então, para qualquer nó i, $\sum_{\ell=1}^{\bar{\ell}(i)} |T_i^\ell|$ tem-se o número mínimo de nós necessários para estabelecer o requisito de k-conectividade do nó i.

A formulação do programa inteiro definido pela função objetivo (4.1) e as restrições (4.2)–(4.11) são uma formulação válida para o problema MIMAX usando o modelo de interferência do receptor (RIM).

$$\text{minimize} \quad Z = INT \tag{4.1}$$

subject to

$$INT \geq \mathrm{I}(i), \forall i \in V \tag{4.2}$$

$$\mathrm{I}(i) \geq \sum_{j \in V \setminus \{i\}} x_j^\ell, \forall i \in V, \ell : i \in T_j^m \tag{4.3}$$

$$\sum_{j \in V} f_{ji}^c - \sum_{l \in V} f_{il}^c = D_c(i), \forall c \in C, \forall i \in V \tag{4.4}$$

$$\sum_{j \in V} f_{ij}^c \leq 1, \forall c \in C, \forall i \in V : i \neq o(c), i \neq d(c) \tag{4.5}$$

$$x_i^\ell \geq f_{ij}^c + f_{ji}^c, \forall i \in V, \forall c \in C, \forall j \in T_i^\ell, \ell = 1, \ldots, \phi(i) \tag{4.6}$$

$$x_i^{\ell+1} \leq x_i^\ell, \forall i \in V, \ell = 1, \ldots, \phi(i) - 1 \tag{4.7}$$

$$x_i^\ell = 1, \forall i \in V, \ell = 1, \ldots, \bar{\ell}(i) \tag{4.8}$$

$$f_{ij}^c \in \{0, 1\}, \forall i, j \in V, \forall c \in C \tag{4.9}$$

$$x_i^\ell \in \{0, 1\}, \forall i \in V, \ell = 1, \ldots, \phi(i) \tag{4.10}$$

$$0 \leq INT \leq |V| \tag{4.11}$$

As restrições (4.2) e (4.3) fornecem a máxima interferência que é minimizada pela função objetivo (4.1). A restrição (4.4) tem como objetivo a conservação de fluxo. A desigualdade (4.5) garante a desconexão dos nós. A desigualdade (4.6) afirma que x_i^ℓ deve ser setado em um se houver um nó $j \in T_i^\ell$ tal que o arco (i, j) ou o arco (j, i) é usado para comunicação do nó i a j pela comodidade c. Ou seja, a desigualdade (4.6) garante que a aresta bidirecional $[i, j]$ seja usada se houver fluxo de i para j ou de j para i.

A restrição (4.7) obriga $x_i^{\ell+1}$ a ser igual a zero se o incremento anterior não for usado, ou seja, se $x_i^\ell = 0$. A restrição (4.8) seta em um as potências incrementais necessárias para atingir pelo menos os k nós mais próximos de cada nó i. As restrições (4.9), (4.10) e (4.11) expressam os requisitos de integralidade e limites das variáveis.

Seja $P_{max} = \sum_{i \in V} \max(p_i)$ o consumo de potência total máximo na rede quando todos os nós transmitem com potência máxima. Uma vez que a função objetivo (4.1) fornece um valor inteiro, pode-se adicionar o consumo total de energia como um valor normalizado no intervalo $[0.0, 1.0)$ sem afetar o cálculo da interferência mínima. Assim, se substituirmos a equação (4.1) pela equação (4.12), tem-se a formulação do programa inteiro para o problema k-conexo MIMAXP usando o modelo RIM.

$$\text{minimize} \quad Z = INT + \left(\frac{\sum_{i \in V} \sum_{\ell=1}^{\phi(i)} q_i^\ell \cdot x_i^\ell}{P_{max}} \right). \tag{4.12}$$

A formulação definida pela função objetivo (4.13) e pelas restrições (4.3)–(4.11) é uma formulação válida para o problema de k-conexo MITOT usando o modelo RIM.

$$\text{minimize} \quad Z = \sum_{i \in V} \mathrm{I}(i) \tag{4.13}$$

Substituindo-se a equação (4.13) pela equação (4.14), tem-se a formulação para o problema k-conexo MITOTP usando, novamente, o modelo RIM.

$$\text{minimize} \quad Z = \sum_{i \in V} \mathrm{I}(i) + \left(\frac{\sum_{i \in V} \sum_{\ell=1}^{\phi(i)} q_i^\ell \cdot x_i^\ell}{P_{max}} \right). \tag{4.14}$$

5 Heurísticas aplicadas à RSSF ($k = 2$)

No capítulo 4 foram apresentadas 5 formulações matemáticas para resolução do problema de redução de consumo de potência e minimização de interferência em RSSF. Sabe-se que a resolução de uma formulação matemática de programação inteira mista possui algumas dificuldades. Uma delas é que usando o *solver* CPLEX só é possível encontrar resultados em tempo computacionalmente aceitável se a RSSF possuir até 50 nós (MORAES; RIBEIRO, 2013).

Uma característica das RSSF é possuir centenas ou milhares de nós, como discutido no capítulo 2. Então a resolução exata se torna limitada para tratar redes maiores (realísticas). Por isso, neste trabalho, são propostas metaheurísticas para reduzir o consumo de potência e reduzir a interferência em RSSF. Neste capítulo são apresentadas as metaheurísticas GRASP, VNS e GRASP-VNS. Também são explicados o algoritmo guloso randomizado e a busca local, que serão usados na implementação das metaheurísticas.

5.1 Greedy Randomized Adaptive Search Procedure - GRASP

Para a resolução de problemas de grande porte, na qual a utilização de um método exato não é adequada devido ao tempo de processamento, utilizam-se heurísticas. Então, pode-se contar com as metaheurísticas, que são métodos de busca inteligente que se adaptam a diversos tipos de problemas e permitem obter uma solução aproximada na presença de restrições de tempo (HÖRNER et al., 2009).

Como afirmado anteriormente, é um processo composto de duas fases: construção da solução inicial e busca local. Na fase de construção, uma solução inicial viável é formada utilizando o algoritmo a ser apresentado na subseção 5.1.2. Na fase de busca local, procura-se melhorar a qualidade da solução inicial. A busca local será apresentada na subseção 5.1.3.

5.1.1 Algoritmos gulosos

Nesta seção, é apresentado o desenvolvimento do algoritmo guloso utilizado na fase de construção da solução inicial para resolução dos problemas MPT, MIMAX, MITOT, MIMAXP e MITOTP.

Os algoritmos gulosos podem ser utilizados para se obter uma solução rápida, pois sempre escolhem o melhor elemento (de acordo com uma função gulosa) a ser incorporado à solução naquele instante para se chegar a um resultado final aceitável. Neste algoritmo, uma escolha feita nunca é revista. São algoritmos determinísticos, ou seja, dado um conjunto de entrada, sempre produzem a mesma saída (BASTOS, 2004). A vantagem dos algoritmos gulosos é o reduzido tempo computacional. Em compensação, não é possível afirmar que a solução obtida é ótima.

Em cenários reais, os enlaces de transmissão das RSSF são assimétricos. Estas arestas podem ter um custo não-negativo $c(u, v)$ associado a elas. Se esta RSSF é assimétrica, o custo de transmissão $c(u, v)$ pode ser diferente do custo $c(v, u)$.

A forma de escolha do melhor elemento que será incorporado à solução é através da avaliação dos elementos por uma função de avaliação gulosa. Para isso, a seguinte função gulosa foi definida.

Definição 1 (função gulosa): o custo estático da aresta (u, v) é dado pelo maior custo dos arcos (u, v) ou (v, u) que a compõem:

$$c(u, v) = \begin{cases} c(u, v), & \text{se } c(u, v) \geq c(v, u); \\ c(v, u), & \text{caso contrário.} \end{cases} \tag{5.1}$$

Desta forma, a rede pode ser considerada simétrica, permitindo usar o algoritmo guloso randomizado apresentado na subseção 5.1.2 para obtenção de uma solução inicial viável. A função gulosa permite que os custos das arestas sejam escolhidos no início.

5.1.2 Algoritmo Guloso Randomizado

O algoritmo receberá como entrada um grafo $G(V, E)$ com entradas assimétricas e a função gulosa, explicada na Seção 5.1.1, realiza o cálculo previsto.

Para ilustrar o desenvolvimento do algoritmo guloso randomizado, apresenta-se o pseudocódigo no Algoritmo 7.

5.1.3 Busca Local

O algoritmo Guloso Randomizado obtém soluções viáveis para todos os problemas estudados: MPT, MIMAX, MITOT, MIMAXP e MITOTP. Uma solução é considerada viável quando atende todos os requisitos para o funcionamento da rede, isto é, ser biconexa bidirecional. Mas as soluções geradas pelo algoritmo guloso randomizado não são necessariamente as soluções ótimas. Então, utiliza-se algoritmos de busca local para melhorar as soluções encontradas pelo algoritmo guloso randomizado e aumentar as chances de se encontrar a solução ótima.

Uma estratégia de busca local do tipo primeiro aprimorante é proposta a seguir, na qual são usadas duas operações básicas para decremento e incremento da potência de um nó qualquer. As operações se baseiam na quantidade de nós afetados em uma operação e no ajuste de potência dos nós afetados. Para a explicação das duas operações,

Algoritmo 7: Algoritmo guloso randomizado

Entrada: $G = (V, E)$, onde V é o conjunto de nós e E é o conjunto de arestas e parâmetro α;
$p_u \leftarrow 0, \forall u \in V$;
$E_c \leftarrow E$;
$E_f \leftarrow 0$;
$V' \leftarrow 0$;
$V' \leftarrow V' \cup \{u\}$;
while $V' \neq V$ **do**
    $LC \leftarrow$ ordenação de E_c por custo;
    $c_{min} \leftarrow$ custo da menor aresta de LC;
    $c_{max} \leftarrow$ custo da maior aresta de LC;
    $lim = c_{min} + \alpha(c_{max} - c_{min})$;
    $LRC \leftarrow (u, v), \forall (u, v) \in LC$ tal que $c(u, v) \leq lim, u \in V', v \notin V'$;
    seleciona-se uma aresta aleatória (u, v) de LRC;
    if (u, v) *é aresta válida* **then**
        remover (u, v) de E_c;
        adicionar (u, v) a E_f;
        $V' \leftarrow \{v\}$;
        atualizar p_u e p_v;
    end if
end while
Saída: Atribuição de potências p e $G(p) = (V, E(p))$;

utilizam-se os conceitos de uma lista ordenada P_i, dos conjuntos S_i^l e os conjuntos T_i^l, apresentados no Capítulo 4, para cada nó $i \in V$ e para cada nível de potência l. A lista P_i contém, em ordem crescente, o conjunto de todos os valores de potência que o nó i pode ter. O conjunto S_i^l contém todos os nós alcançados com o nível de potência $p_i^l \in P_i$. $T_i^l = S_i^l - S_i^{l-1}$ é o conjunto de nós alcançados com o aumento de potência de p_i^{l-1} para p_i^l. Se $l = 1$, tem-se que $T_i^l = S_i^l$. Sabendo que toda busca local se inicia com uma solução viável, então assume-se que a potência p_i de cada nó $i \in V$ é tal que $p_i = p_i^l$, para algum $l \in 1, ..., x(i)$. Usando a Figura 13 como exemplo, os valores de T_A^l estão representados na legenda. Se aumentar a potência do nó A de dois para três unidades de potência, o nó A passa a alcançar os nós C e F, além do nó D que já estava sendo alcançado (MORAES; RIBEIRO; DUHAMEL, 2009).

A Figura 14 exemplifica arco e aresta. Uma aresta $\{i, j\}$, ou seja $\{i, j\} = \{j, i\}$ é a união de dois arcos unidirecionais ordenados $(i, j) \neq (j, i)$ tal que (i, j) significa $i \rightarrow j$ (i pode estabelecer comunicação direta com j) e (j, i) significa $j \rightarrow i$ (j pode estabelecer comunicação direta com i). Assim $[i, j] = (j, i) \cup (i, j)$, ou seja $\{i, j\} \Rightarrow i \leftrightarrow j$. As setas em vermelho mostram as arestas no grafo direcionado. Já as setas pretas representam os arcos.

Também foi definida uma lista $A_i = (a_i^1, ..., a_i^n)$, para todo nó $i \in V$ com $a_i^l \in 0, 1, 2$

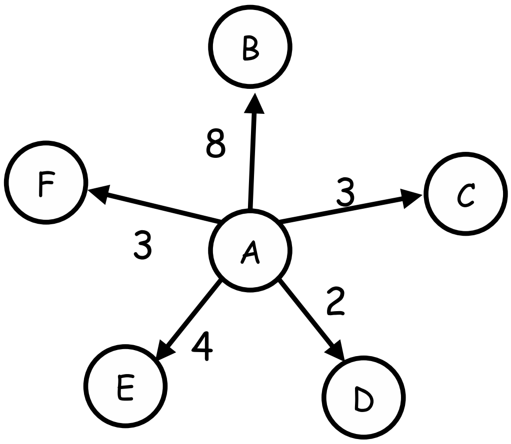


Figura 13 – $Q_A = [2, 1, 1, 4]$, $T_A^1 = [D]$, $T_A^2 = [C, F]$, $T_A^3 = [E]$ e $T_A^4 = [B]$

Adaptado de: (MORAES, 2009)

para $l \in 1, ..., x(i)$. Se $a_i^l = 0$, o nível de potência necessário para estabelecer comunicação entre os nós é maior do que a potência corrente do nó. Se $a_i^l = 1$, a potência corrente do nó é suficiente para estabelecer comunicação de um nó i para o nó j, mas não é suficiente para estabelecer comunicação no sentido de j para i, ou seja, o nível de potência do nó i não suporta o arco (i, j), somente a aresta $\{i, j\}$. Se $a_i^l = 2$, a potência corrente do nó é suficiente para estabelecer comunicação entre o nó i e j em ambas direções, ou seja, o nível de potência do nó i suporta o arco (i, j) (MORAES; RIBEIRO; DUHAMEL, 2009).

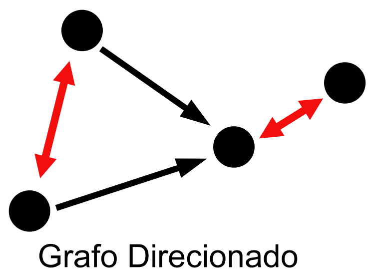


Figura 14 – Exemplo de arco e aresta em um grafo direcionado

Fonte: (MORAES, 2009)

A Figura 15 mostra um exemplo de conjuntos P_i e A_i para uma rede de sensores. Neste exemplo, a potência do nó i é 8, de modo que o nó i consegue se conectar com todos os nós que estão a sua volta. Como a potência do nó k, por exemplo, é 3 e o custo $c(k, i)$ é 3, também há estabelecimento de *link* de comunicação de i para k e de k para i, por isso $a_i^l = 2$ para todos os 4 quatro nós a sua volta. Neste mesmo exemplo, a potência do nó j é 7. Como a potência do nó k, por exemplo, é 3 e o custo $c(j, k)$ é 4, há estabelecimento de *link* de comunicação de j para k, mas não há de k para j, por isso $a_j^l = 1$. Neste mesmo exemplo, $a_k^l = 0$.

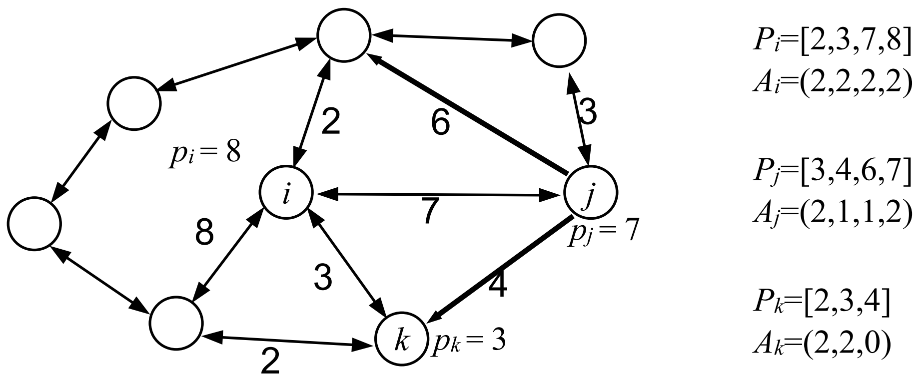


Figura 15 – Exemplo dos conjuntos P_i e A_i para uma solução viável

Fonte: (MORAES, 2009)

A operação de decremento (p_i, i) é definida de forma que, a partir de uma potência de transmissão p_i e de um nó i, decrementa-se o nível de potência do nó i de p_i para p_i^-, sendo que p_i^- é o nível mais alto de potência menor que p_i a suportar uma aresta. Nesta operação, é realizada a remoção de, pelo menos, um arco da solução corrente. Na operação de incremento (p_i, i), a potência do nó i é aumentada de p_i para p_i^+. Nesta operação, há acréscimo de pelo menos uma aresta na solução corrente. A inclusão de arestas é mais eficiente que a inclusão de arcos, pois somente a inclusão de aresta pode ajudar na recuperação da viabilidade de uma solução (biconexidade) (MORAES, 2009).

Na busca local, aplica-se diversas operações de decremento em uma solução viável até provocar a quebra de sua viabilidade, criando uma solução inviável. A inviabilidade ocorre devido a perda de biconexidade. Após, são realizadas diversas operações de incremento à solução inviável a fim de recuperar a sua viabilidade ou até que o custo desta nova solução seja maior que o custo da solução corrente. As operações de decremento são realizadas antes das operações de incremento, porque se as operações de incremento fossem realizadas primeiro, não haveria quebra de viabilidade de solução.

Na Figura 16, tem-se no primeiro quadro um exemplo de uma solução corrente biconexa para uma RSSF. Também é representado no exemplo os conjuntos P_i e A_i para os nós i, j e k. No segundo quadro, mostra-se que foi realizada uma operação de decremento no nó j, fazendo $p_j = 7$ ir para $p_j^- = 6$. Com esta operação, há a perda do arco(j,i). É realizada a segunda operação de decremento no nó j, fazendo com que $p_j = 6$ vá para $p_j^- = 4$. Com isso, há a perda do arco$(j, x) = 6$. Mais uma operação de decremento é realizada no nó j, então $p_j = 4$ vai para $p_j^- = 3$. Nesta operação, há a perda do arco(j, k). Com isso, perde-se a viabilidade da solução. No terceiro quadro, é realizada a operação de incremento para recuperação da viabilidade da solução (biconexidade). Então, o nó k tem sua potência aumentada de $p_k = 3$ para $p_k^- = 4$. Como explicado anteriormente, na operação de incremento há acréscimo de aresta. Neste exemplo, acrescenta-se a aresta

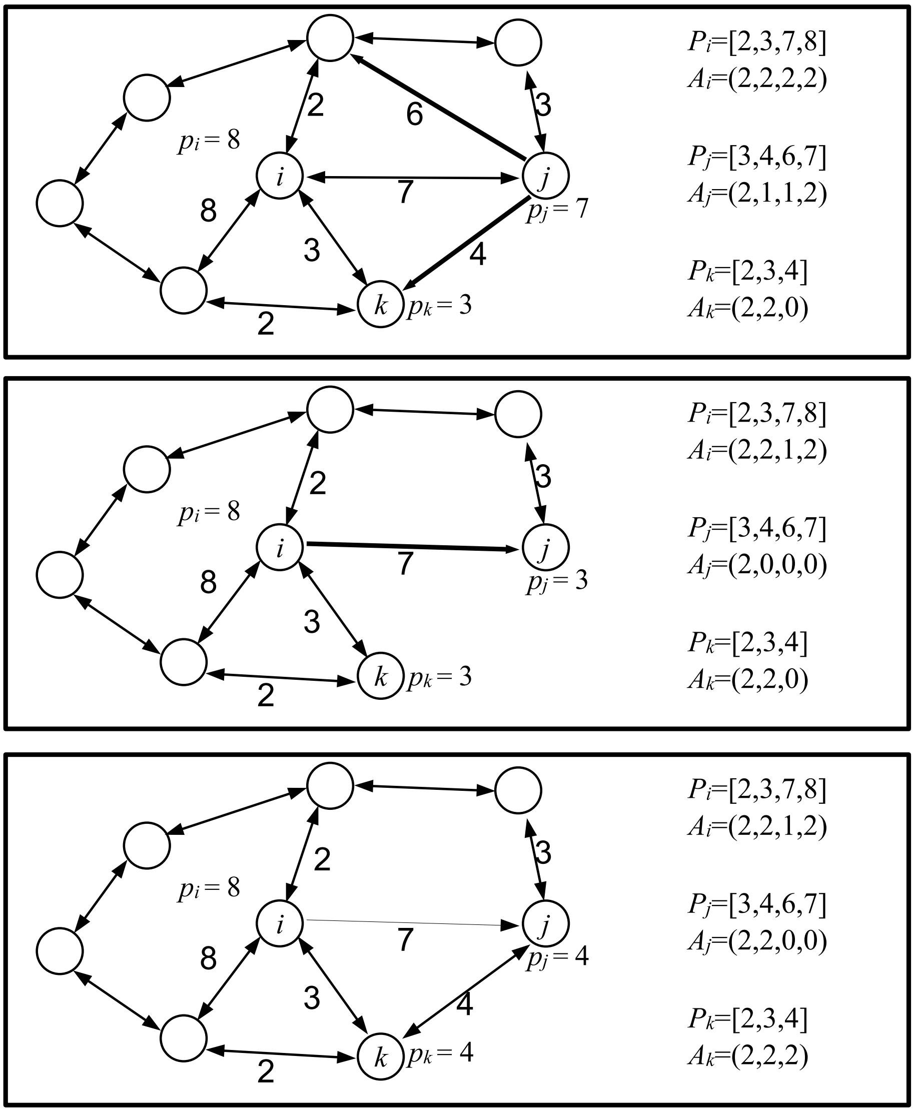


Figura 16 – Exemplo das operações de decremento e incremento

Fonte: (MORAES, 2009)

{j,k}. Assim $p_j = 4$ e $p_k = 4$.

Para ilustrar a implementação da Busca Local, apresenta-se o pseudocódigo no Algoritmo 8. Como entrada, a busca local recebe uma solução inicial viável s, formada pela atribuição de potência p e $G(p)$. Na linha 2, é criada uma variável booleana que recebe valor verdadeiro. Na linha 6, é calculado o decremento de cada nó u da rede, ou seja, quanto cada nó pode reduzir a potência. Na linha 7, esses cálculos são colocados em ordem decrescente em uma lista. No laço das linhas 8 a 16, a potência de um nó é alterada de acordo com a lista de decremento criada. Remove-se sempre a primeira opção de decremento da lista e testa-se a biconexidade do grafo. As potências serão alteradas de acordo com essa lista de decremento até gerar uma solução inviável. É criada uma lista de incremento de potência para os nós da rede e coloca-se esta lista em ordem crescente (linhas 27 e 28). No laço das linhas 26 a 32, as potências são aumentadas de acordo com a lista de incremento, até que uma nova solução viável seja encontrada.

Para ilustrar a implementação da heurística GRASP, explicada acima, apresenta-se o Algoritmo 9 que mostra o pseudocódigo do GRASP.

Na linha 2, o melhor valor da função objetivo é inicializado com infinito, porque ainda não há solução inicial criada e como o objetivo é reduzir potência e interferência, é necessário usar um valor bem alto para ir reduzindo. O laço das linhas 3 a 10 é repetido até que o critério de parada seja alcançado. Na linha 4, uma solução é construída através do Algoritmo 7. Na linha 5, executa-se o Algoritmo 8. Na linha 7, testa-se se houve melhoria na solução. Se sim, então a solução corrente é substituída pela melhor solução, já que a busca local usada é do tipo primeiro aprimorante.

5.2 Variable Neighborhood Search - VNS

É uma metaheurística proposta por Mladenovic [1995], que explora vizinhanças mais distantes da solução atual. A busca é feita através de transições no espaço de busca e é bastante robusta. As principais características do VNS são: exploração de vizinhanças maiores e mudança sistemática de vizinhança. O VNS difere de algumas metaheurísticas com relação a estratégia usada para percorrer o espaço de busca. Algumas metaheurísticas aceitam a degradação da solução atual para sair de um ótimo local. O VNS altera a vizinhança para fugir de um ótimo local. O VNS é fundamentado em três observações importantes (MLADENOVIC, 1995):

- Um mínimo local de uma estrutura de vizinhança pode ser diferente do mínimo local de outra estrutura de vizinhança;
- Um mínimo global é um mínimo local em relação a todas as estruturas de vizinhança;
- Os ótimos locais para uma ou mais estruturas de vizinhança apresentam estrutura parecida. Este fato mostra que um ótimo local fornece informações sobre o ótimo global.

No VNS básico (BVNS), tem-se as etapas de pertubação da solução corrente, uma busca local e mudança de vizinhança. Geralmente, a busca utilizada no BVNS é do tipo primeiro aprimorante. Para ilustrar o desenvolvimento do VNS básico, apresenta-se o Algoritmo 10, que mostra o pseudocódigo do BVNS.

O algoritmo inicia recebendo uma solução inicial s, o número de estruturas de vizinhanças k_{max} e o tempo de execução t_{max}. No laço das linhas 3 a 10, a função *Shake* (linha 6) realiza movimentos aleatórios na solução s dentro da vizinhança k e gera uma nova solução s'.

Algoritmo 8: Busca local

```
Entrada: Atribuição de potências p e G(p) (solução s);
condicao ← verdadeiro
while condição do
    while condição1 do
        condicao1 ← verdadeiro
        calcular valor de decremento de cada nó u ∈ V;
        gerar lista de decremento em ordem decrescente;
        while lista ≠ 0 do
            alterar potências p de acordo com lista criada;
            remove primeira opção da lista de decremento;
            testar a biconexidade do grafo;
            if G(p) é biconexo then
                condicao1 ← falso
                condicao ← falso
                else
                    p ← p'
                end if
            end if
        end while
    end while
    p' ← p
    calcular valor de decremento de cada nó u ∈ V;
    gerar lista de decremento em ordem decrescente;
    while lista ≠ 0 do
        alterar potências de acordo com lista criada;
        remove primeira opção da lista;
        calcular valor de incremento de cada nó u ∈ V;
        gerar lista de incremento em ordem crescente;
        testar a biconexidade do grafo;
        while G não é biconexo OU pot. tot. da nova solução (s'') não é melhor do
          que pot. tot. da solução corrente(s) do
            alterar potências de acordo com lista de incremento criada;
            remove primeira opção da lista de incremento;
            if pot. tot. da nova solução (s'') é melhor do que pot. tot. da solução
              corrente(s) then
                p' ← p
                condicao ← falso
                else
                    p ← p'
                end if
            end if
        end while
    end while
end while
Saída: Atribuição de potências p' (solução s'')
```

Algoritmo 9: Algoritmo GRASP

Entrada: Grafo $G = (V, E)$, parâmetro α, critério de parada;
$f(s*) \leftarrow \infty$
for *critério de parada não satisfeito* **do**
    $s \leftarrow$ `Algoritmo_Guloso_Randomizado`(G,α)
    $s'' \leftarrow$ `BuscaLocal`(s)
    if *f(s") < f(s*)* **then**
        $f(s*) \leftarrow f(s'')$
    end if
end for
Saída: $f(s*)$

Algoritmo 10: VNS básico

Entrada: s (solução inicial), k_{max} (número de estruturas de vizinhança), t_{max} (tempo máximo)
Saída: s (solução)
for $tempo \leq t_{max}$ **do**
    $k \leftarrow 1$
    for $k \leq k_{max}$ **do**
        $s' \leftarrow shake(s, k)$
        $s'' \leftarrow BuscaLocal(s')$
        $s \leftarrow mudancavizinhanca(s, s'', k)$
        $tempo \leftarrow tempoalgoritmo()$
    end for
end for

Um pseudocódigo da função *Shake* pode ser visto no Algoritmo 11. Este algoritmo recebe como entrada a solução s e o valor de k. Na linha 2, o algoritmo testa se $k = 1$. Se sim, será escolhido um nó aleatório e é realizada a redução de potência desse nó. Por exemplo, se um nó i possui uma potência $p = 7$ e sua lista $P_i = 2, 3, 4, 7$, então a potência p de i será igual a 4. A redução de potência é realizada em nós nos quais é possível reduzir a potência e manter a viabilidade da solução. Na linha 5, o algoritmo testa se $k = 2$. Se sim, será escolhido um nó aleatório e é realizado o aumento de potência desse nó.

Algoritmo 11: Função shake - VNS

1 **Entrada:** s e k (vizinhança atual);
2 **if** $k = 1$ **then**
3   escolher um nó aleatório de s para reduzir potência mantendo solução viável
4 **end if**
5 **if** $k = 2$ **then**
6   escolher um nó aleatório de s para aumentar potência
7 **end if**
8 **Saída:** s'

Na linha 7, a função $BuscaLocal$ recebe como entrada a solução s' e gera uma nova solução s''. A solução s'' é uma solução melhor do que a solução s', obtida no Algoritmo 11. A busca local é do tipo primeiro aprimorante.

Após, é executada a função $MudançaVizinhança$, na linha 8. Um pseudocódigo da função $MudancaVizinhanca$ pode ser visto no Algoritmo 12. Neste algoritmo, é avaliado se a solução s'' obtida pela busca local é melhor do que a solução corrente, s. Isso é realizado na linha 2. Se a solução s'' for melhor, esta se torna a nova solução corrente na linha 3. Nesse caso, k permanece igual a 1, na linha 4, para que seja feita uma nova busca de solução melhor a partir desta nova solução. Se a solução s'' não for melhor do que s na linha 4, então são explorados novos espaços de busca alterando o valor de k, como mostrado na linha 6 do algoritmo.

O BVNS termina quando atinge o tempo de processamento pré-definido. Este tempo pode estar relacionado com o tempo de execução, como na linha 9 do Algoritmo 10, tempo sem melhoria, etc.

Algoritmo 12: Função Mudança de vizinhança - VNS

1 **Entrada:** s, s', k
2 **if** *f(s")* $<$ *f(s)* **then**
3     $s \leftarrow s''$
4     $k \leftarrow 1$
5 **end if**
6 $k \leftarrow k + 1$
7 **Saída:** s

5.3 Algoritmo Híbrido

Nesta metaheurística, o algoritmo GRASP é combinado com a estratégia de busca do VNS. Estes algoritmos foram combinados, pois o GRASP fortalece a diversificação das soluções. Já o algoritmo VNS incentiva a intensificação da solução (LÓPEZ-SÁNCHEZ; SÁNCHEZ-ORO; HERNÁNDEZ-DÍAZ, 2019).

O algoritmo GRASP com VNS é construído utilizando o Algoritmo 10 como fase de busca local na linha 5 do Algoritmo 9. Assim, obtém-se o Algoritmo 13.

Algoritmo 13: Algoritmo Híbrido

Entrada: Grafo $G = (V, E)$, parâmetro α, critério de parada, k_{max} (número de estruturas de vizinhança);
$f(s*) \leftarrow \infty$
for *critério de parada não satisfeito* **do**
    $s \leftarrow$ `Algoritmo_Guloso_Randomizado`(G,α)
    $s'' \leftarrow$ `VNSmodificado`(s)
    if *f(s”) < f(s*)* **then**
        $f(s*) \leftarrow f(s'')$
    end if
end for
Saída: $f(s*)$

6 Análise de Resultados das simulações

Neste capítulo são apresentadas as análises dos resultados das simulações para o problema FI/SMEMIT na seção 6.1. Na seção 6.2 são apresentadas as análises para os resultados dos problemas MPT, MIMAX, MITOT, MIMAXP e MITOTP, sendo que são discutidos o comportamento das heurísticas para cada um destes problemas, são comparados os resultados das metaheurísticas com resultados ótimos e são discutidos os resultados obtidos com o algoritmo Híbrido para instâncias realísticas.

6.1 Problema FI/SMEMIT

Nesta seção são apresentados os resultados para o problema MPT obtido com os algoritmos de custo de envio, algoritmo de cobertura, algoritmo FI/SMEMIT, heurísticas GRASP e VND. Os resultados são discutidos e analisados no sentido de melhor atender as necessidades de uma rede de sensores sem fio.

As simulações computacionais foram realizadas utilizando um conjunto de instâncias geradas aleatoriamente, com |V| de 20 a 300 nós. Para cada |V|, foram geradas 5 instâncias diferentes. Nesta classe, as instâncias foram geradas em uma grade unitária. Os nós foram posicionados aleatoriamente na grade. O custo das arestas $c(u, v)$ é resultado de uma perturbação aleatória F, com $F \in [0, 8; 1, 2]$, e o quadrado da distância euclidiana d_{uv}:

$$c(u, v) = F.d^2_{u,v} \tag{6.1}$$

Os algoritmos foram implementados no software Matlab 2017 em uma máquina com core i5, 8Gb de memória RAM e 2.7GHz. Nas tabelas, as colunas estão relacionadas aos algoritmos descritos no Capítulo 3, sendo o Guloso 1 referente ao Algoritmo 3.1, Guloso 2 referente ao Algoritmo 3.2, FI/SMEMTI referente ao Algoritmo 3.4, o GRASP referente ao Algoritmo 3.5 e VND referente ao Algoritmo 3.6.

Os valores médios de interferência máxima obtidos, utilizando o modelo RIM, são mostrados na Tabela 1. Os resultados mostram a média das cinco execuções para cada valor de |V|. Os algoritmos gulosos tiveram os piores resultados para interferência máxima. Os resultados do Guloso 1 ficaram, em média, 21,68% acima dos resultados do VND. Já os resultados do Guloso 2, ficaram 15,47% acima. Os resultados do FI/SMEMIT adaptado de (PANDA; SHETTY, 2012) ficaram, em média, 7,15% acima do VND. O GRASP e o VND obtiveram melhores resultados, sendo que o GRASP alcançou valores iguais ao VND em |V| igual a 30, 40 e 250 nós, mas os resultados, em geral, ficaram 4,07% acima dos resultados do VND. Para uma melhor comparação, a Figura 3 mostra um gráfico |V| x

Imax onde foram plotados os valores da Tabela 1.

Tabela 1 – Valores médios absolutos e relativos de interferências máximas para instância EU

	Guloso1		Guloso2		FI/SMEMIT		GRASP		VND	
|V|	Abs.	Erro(%)	Abs.	Erro(%)	Abs.	Erro(%)	Abs.	Erro(%)	Abs.	Erro(%)
20	3,20	25,00	3,00	20,00	2,00	14,29	2,80	14,29	**2,40**	**0,00**
30	3,40	17,65	3,20	12,50	3,00	6,67	2,80	0,00	**2,80**	**0,00**
40	4,00	25,00	3,60	16,67	3,20	6,25	3,00	0,00	**3,00**	**0,00**
100	4,80	29,17	4,20	19,05	3,60	5,56	3,60	5,56	**3,40**	**0,00**
150	5,40	25,93	4,80	16,67	4,40	9,10	4,20	4,76	**4,00**	**0,00**
200	5,80	20,69	5,40	14,81	5,00	8,00	4,80	4,17	**4,60**	**0,00**
250	5,60	10,71	5,40	7,41	5,00	0,00	5,00	0,00	**5,00**	**0,00**
300	6,20	19,35	6,00	16,67	5,40	7,41	5,20	3,85	**5,00**	**0,00**
Geral	4,80	21,68	4,45	15,47	4,05	7,15	3,93	4,07	**3,78**	**0,00**

Os valores de potências totais obtidos são mostrados na Tabela 2. O princípio do Algoritmo guloso sobre custo de envio (Guloso 1) é conectar os menores enlaces, então é aceitável que as suas potências totais sejam as menores possíveis. Ou seja, o Guloso 1 obteve os melhores resultados para a potência total. Os resultados do Guloso 2 ficaram, em média, 2,17% acima dos resultados do Guloso 1. Os resultados do FI/SMEMIT adaptado de (PANDA; SHETTY, 2012) ficaram, em média, 1,54% acima do Guloso 1. GRASP e o VND obtiveram bons resultados comparando-os com o Guloso 1. Além disso, seus resultados ficaram, em média, 1,01% acima dos resultados do Guloso 1. Apesar dos valores de potência total alcançados pelo VND não serem os menores, os resultados ficaram, em geral, 0,59% acima dos resultados do Guloso 1.

Tabela 2 – Valores médios absolutos e relativos de potências totais para instância EU

	Guloso1		Guloso2		FI/SMEMIT		GRASP		VND	
|V|	Abs.	Erro(%)	Abs.	Erro(%)	Abs.	Erro(%)	Abs.	Erro(%)	Abs.	Erro(%)
20	**1,75**	**0,00**	1,79	2,28	1,81	3,42	1,77	1,14	1,77	1,14
30	**1,84**	**0,00**	1,87	1,63	1,86	1,08	1,86	1,08	1,85	0,54
40	**1,78**	**0,00**	1,82	2,24	1,81	1,68	1,79	0,56	1,79	0,56
100	**1,87**	**0,00**	1,92	2,67	1,89	1,07	1,89	1,07	1,88	0,53
150	**1,98**	**0,00**	2,02	2,02	2,01	1,51	2,01	1,51	1,99	0,50
200	**2,24**	**0,00**	2,27	1,34	2,27	1,34	2,26	0,89	2,24	0,00
250	**2,59**	**0,00**	2,66	2,70	2,62	1,15	2,62	1,15	2,62	1,15
300	**2,84**	**0,00**	2,91	2,46	2,87	1,05	2,86	0,70	2,85	0,35
Geral	**2,11**	**0,00**	2,15	2,17	2,14	1,54	2,13	1,01	2,12	0,59

6.2 Problemas MPT, MIMAX, MIMAXP, MITOT e MITOTP

Nesta seção são apresentados os resultados para os problemas MPT, MIMAX, MITOT, MIMAXP e MITOTP obtidos com os algoritmos/heurísticas GRASP, VNS e o Híbrido. Os resultados são discutidos e analisados no sentindo de melhor atender as necessidades de uma rede de sensores sem fio.

Todos os algoritmos foram desenvolvidos e compilados no *software* Matlab, versão

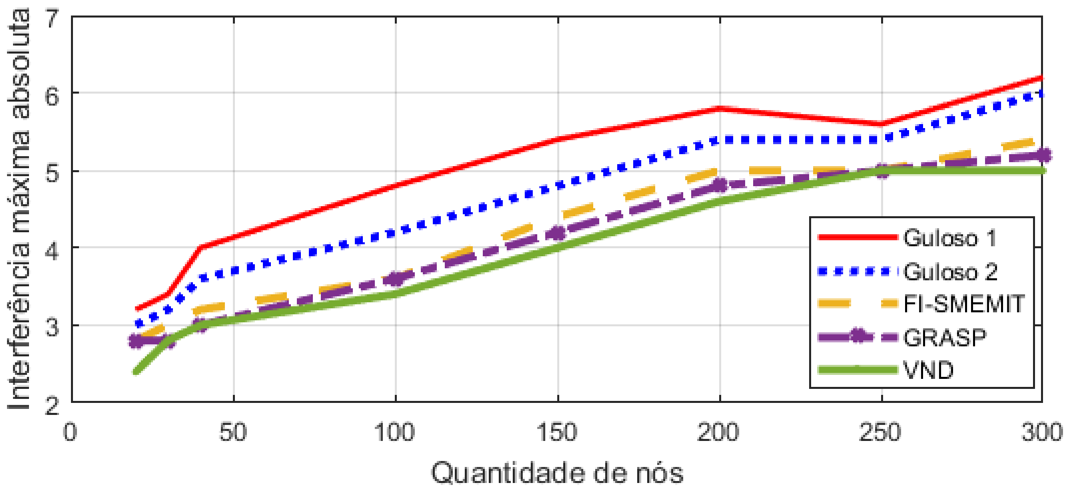


Figura 17 – Resultados de $I_{max}x|V|$ em valores absolutos

2018a. Os experimentos foram realizados em uma máquina Intel Core I5-8265 com relógio de 1,8 GHz e 8 Gb de memória RAM com sistema operacional Windows 10 Home.

Os experimentos foram realizados com três classes de instâncias assimétricas geradas randomicamente, sendo:

- Instâncias euclidianas, denominadas EU;
- Instâncias euclidianas, denominadas DE;
- Instâncias randômicas, denominadas RD.

Primeiramente, foi gerada uma classe de instâncias euclidianas (EU). São instâncias assimétricas, que foram geradas randomicamente, na qual $|V| \in [10, 800]$ nós. Nesta classe, as instâncias foram geradas em uma grade unitária. Os nós foram posicionados aleatoriamente na grade. O custo das arestas $c(u, v)$ é resultado de uma perturbação aleatória F, com $F \in [0, 8; 1, 2]$, e o quadrado da distância euclidiana d_{uv}:

$$c(u, v) = F.d^2_{u,v} \tag{6.2}$$

As instâncias DE foram geradas de maneira similar as instâncias EU, utilizando a equação 6.2 para gerar os custos das arestas. A diferença entre elas é que nas instâncias EU, a grade permanece com lado constante igual a um e nas instâncias DE, a grade tem seu tamanho alterado para que a densidade permaneça constante. Ou seja, nas instâncias EU, o valor dos lados da grade é constante e fixado em $1, 0$. Nas instâncias DE, a densidade se mantém constante e igual a um nó por unidade quadrada. As instâncias RD foram geradas aleatoriamente e os custos da arestas $c(u, v)$ estão compreendidos no intervalo $(0.0000, 1.0000]$.

Antes da apresentação dos resultados obtidos, é necessário mostrar a seleção dos parâmetros utilizados nas metaheurísticas. A primeira análise realizada foi para a escolha do melhor valor para o parâmetro α presente no Algoritmo 7. Como visto no Capítulo 5, α é um valor escolhido entre 0,0 e 1,0. Os valores usados são mostrados na Tabela 3.

Tabela 3 – Parâmetros utilizados nos algoritmos GRASP, VNS e Híbrido

Parâmetros	Descrição	Valor
α	Parâmetro da LRC	0,2
k	Número de vizinhanças	2

De acordo com Gendreau, Potvin et al. [2010], utilizar um α fixo em 0,2 produz boas soluções, pois afirmam que as soluções terão uma qualidade mais próxima das soluções de um algoritmo puramente guloso, além de uma intensificação relativamente maior, comparando com valores de α mais próximos de 1,0, em que o esforço computacional e o aumento da diversidade são maiores, produzindo soluções de qualidade inferior.

Para cada tamanho da rede (relacionada a V), 15 instâncias testes foram geradas aleatoriamente. Todas as instâncias geradas são representadas como grafos completos. Os experimentos consistem em uma execução de cada uma das quinze instâncias EU, DE e RD. Os resultados apresentados são uma média das quinze instâncias. As análises foram feitas para o caso biconexo ($k = 2$).

6.2.1 Comportamento das metaheurísticas GRASP, VNS e Híbrido para o problema MPT

Neste primeiro experimento, cada metaheurística foi executada durante um tempo máximo de 1800 segundos. As tabelas 4, 5 e 6 apresentam os resultados médios das 15 execuções para cada metaheurística para o problema de minimização de potência (MPT) para as instâncias EU25, EU50 e EU100, sendo EU25 uma instância euclidiana com 25 nós, EU50 uma instância euclidiana com 50 nós e EU100 uma instância euclidiana com 100 nós. Nas tabelas, os valores em negrito destacam o melhor resultado dentro de cada tempo estabelecido.

Essas comparações auxiliam na análise dos algoritmos, na qual observa-se qual possui a capacidade de alcançar melhores resultados em menor tempo. Também ajudam a analisar qual algoritmo apresenta melhores soluções para diferentes tamanhos de instâncias.

Os resultados das Tabelas 4 e 5 mostram que, para instâncias menores (EU25 e EU50), as metaheurísticas GRASP, VNS e Híbrida encontram soluções ótimas, mas a metaheurística Híbrida é um pouco mais lenta para alcançar tais resultados. As soluções ótimas estão destacadas em negrito. Isto pode ser explicado pelo maior tempo necessário para percorrer a vizinhança e também devido as maiores modificações ocasionadas a solução atual, já que o GRASP é o algoritmo que alcança melhores resultados em menor tempo.

Tabela 4 – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrida para a instância EU25 aplicada ao problema MPT

Metaheurística	**Tempo (s)**					
	5	**30**	**120**	**300**	**600**	**1800**
GRASP	**1,3588**	**1,3588**	**1,3588**	**1,3588**	**1,3588**	**1,3588**
BVNS	1,3593	**1,3588**	**1,3588**	**1,3588**	**1,3588**	**1,3588**
HÍBRIDO	1.3592	**1,3588**	**1,3588**	**1,3588**	**1,3588**	**1,3588**

Tabela 5 – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrida para a instância EU50 aplicada ao problema MPT

Metaheurística	**Tempo (s)**					
	5	**30**	**120**	**300**	**600**	**1800**
GRASP	**1,2647**	**1,2625**	**1,2603**	**1,2603**	**1,2603**	**1,2603**
BVNS	1,2651	1,2629	**1,2603**	**1,2603**	**1,2603**	**1,2603**
HÍBRIDO	1,2651	1,2629	1,2605	**1,2603**	**1,2603**	**1,2603**

Na Tabela 6, pode-se observar que a metaheurística Híbrida alcança melhores soluções para minimização de potência em menor tempo comparando com as metaheurísticas GRASP e VNS.

Tabela 6 – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrida para a instância EU100 aplicada ao problema MPT

Metaheurística	**Tempo (s)**					
	5	**30**	**120**	**300**	**600**	**1800**
GRASP	1,4427	1,4302	1,4281	1,4179	1,4162	1,4042
BVNS	**1,4425**	1,4286	1,4274	1,4153	**1,3578**	**1,3497**
HÍBRIDO	1,4428	**1,4242**	**1,4133**	**1,3977**	**1,3578**	**1,3497**

As Figuras 18, 19 e 20 mostram os resultados obtidos na execução de cada metaheurística para as instâncias EU200, EU400 e EU800, respectivamente. Tais informações, em forma de tabela, podem ser conferidas no Apêndice B.

Além disso, a metaheurística Híbrida continua melhorando a solução a medida que mais tempo é fornecido. Esta análise é corroborada pelos resultados médios obtidos para as instâncias EU200, EU400 e EU800 e resumida na Tabela 8.

Tabela 7 – Evolução dos resultados do Híbrido para as instâncias EU200, EU 400 para o problema MPT - Tempo acima de 1800s

Tipo	**Tempo (s)**				
	1800	**1950**	**2100**	**2250**	**2400**
EU200	1,7289	1,7154	1,7012	1,6937	1,6937
EU400	2,8802	2,8598	2,8322	2,8273	2,8148
EU800	5,0794	5,0672	5,0589	5,0502	5,0409

Como afirmado anteriormente, as RSSF possuem centenas de nós sensores, ou seja, são consideradas instâncias de grande porte. Para este tipo de instância, a metaheurística Híbrida alcança os melhores resultados médios, ou seja, em instâncias maiores a convergência do Híbrido é mais rápida, como pode ser observado na Tabela 6 e nas Figuras 18,

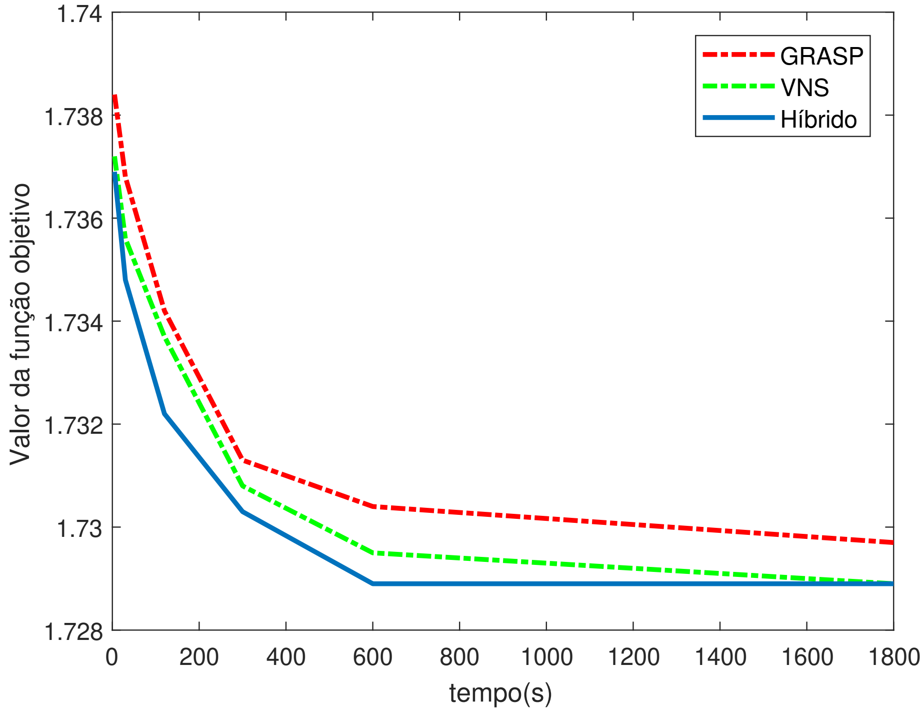


Figura 18 – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrida para a instância EU200

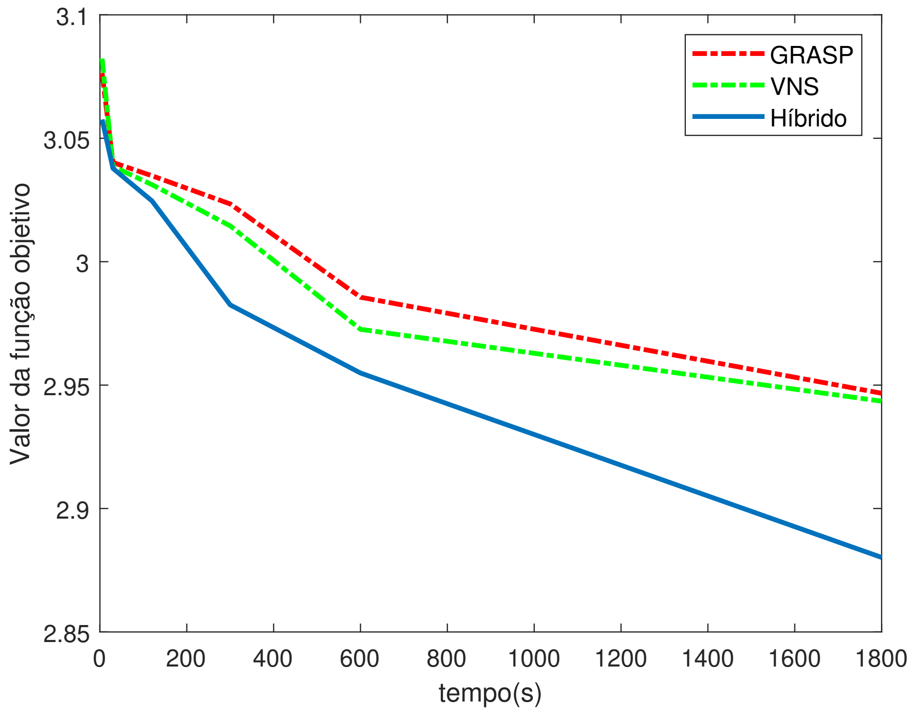


Figura 19 – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrida para a instância EU400

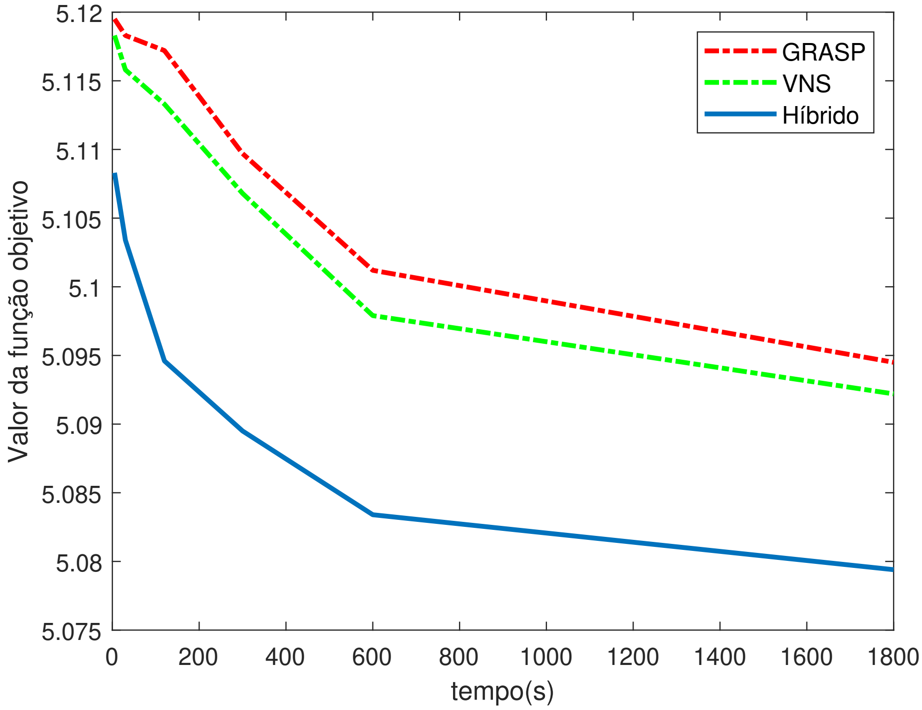


Figura 20 – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrida para a instância EU800

Tabela 8 – Evolução dos resultados do Híbrido para as instâncias EU200, EU 400 e EU800 para o problema MPT

Tipo	**Tempo (s)**				
	1800	**1950**	**2100**	**2250**	**2400**
EU200	1,7369	1,7289	1,7289	1,7012	1,6937
EU400	3,0576	2,9549	2,8802	2,8322	2,8148
EU800	5,1083	5,0834	5,0794	5,0589	5,0409

19 e 20. Na próxima seção, os resultados obtidos com as metaheurísticas para as instâncias EU serão comparados com os resultados ótimos, pois só são conhecidos os valores ótimos para as instâncias EU.

6.2.2 Comportamento das metaheurísticas GRASP, VNS e Híbrido para o problema MIMAX

Neste experimento, cada metaheurística foi executada durante um tempo máximo de 1800 segundos. A tabela 9 apresenta os resultados médios das 15 execuções para cada metaheurística para o problema MIMAX para as instâncias EU100, EU200 e EU400, sendo EU100 uma instância euclidiana com 100 nós, EU200 uma instância euclidiana com 200 nós e EU400 uma instância euclidiana com 400 nós. Na tabela, os valores em negrito destacam o melhor resultado dentro de cada tempo estabelecido.

Pode-se observar que a metaheurística Híbrida alcança melhores soluções em menor tempo comparando com as metaheurísticas GRASP e VNS. Comparando com o problema MPT (mostrado na seção 6.2.1), vê-se que o Híbrido alcança melhores soluções em menor tempo no problema MIMAX, pois no problema MIMAX foram utilizados instâncias maiores (EU maior que 100).

Tabela 9 – Evolução dos resultados do Híbrido para as instâncias EU100, EU 200 e EU400 para o problema MIMAX

	Tempo (s)	GRASP	VNS	Híbrido
$\lvert V\rvert = 100$	30	6,67	**6,57**	**6,57**
	120	6,50	**6,40**	**6,40**
	600	6,47	6,37	**6,33**
	1800	6,40	6,37	**6,33**
$\lvert V\rvert = 200$	30	7,50	**7,40**	**7,40**
	120	7,43	7,37	**7,33**
	600	7,40	7,33	**7,27**
	1800	7,37	7,33	**7,27**
$\lvert V\rvert = 400$	30	7,67	7,53	**7,47**
	120	7,50	7,47	**7,40**
	600	7,43	7,40	**7,33**
	1800	7,40	7,33	**7,27**

6.2.3 Comportamento das metaheurísticas GRASP, VNS e Híbrido para o problema MIMAXP

Neste experimento, cada metaheurística foi executada durante um tempo máximo de 1800 segundos. A tabela 10 apresenta os resultados médios das 15 execuções para cada metaheurística para o problema MIMAXP para as instâncias EU100, EU200 e EU400, sendo EU100 uma instância euclidiana com 100 nós, EU200 uma instância euclidiana com 200 nós e EU400 uma instância euclidiana com 400 nós. Na tabela, são apresentados os valores de potência e de interferência máxima e os valores em negrito destacam o melhor resultado dentro de cada tempo estabelecido.

6.2.4 Comportamento das metaheurísticas GRASP, VNS e Híbrido para o problema MITOT

Neste experimento, cada metaheurística foi executada durante um tempo máximo de 1800 segundos. A tabela 11 apresenta os resultados médios das 15 execuções para cada metaheurística para o problema MITOT para as instâncias EU100, EU200 e EU400, sendo EU100 uma instância euclidiana com 100 nós, EU200 uma instância euclidiana com 200 nós e EU400 uma instância euclidiana com 400 nós. Na tabela, são apresentados os valores de potência e os valores em negrito destacam o melhor resultado dentro de cada tempo estabelecido.

Tabela 10 – Evolução dos resultados do Híbrido para as instâncias EU100, EU 200 e EU400 para o problema MIMAXP

		Potência			Interferência Máxima		
	Tempo(s)	GRASP	VNS	Híbrido	GRASP	VNS	Híbrido
$\|V\| = 100$	30	1,43	**1,42**	**1,42**	6,67	**6,57**	**6,57**
	120	1,42	**1,41**	**1,41**	6,50	**6,40**	**6,40**
	600	1,41	1,40	**1,39**	6,47	6,37	**6,30**
	1800	1,40	**1,39**	**1,39**	6,40	6,37	**6,30**
$\|V\| = 200$	30	1,82	**1,80**	**1,80**	7,50	**7,40**	**7,40**
	120	1,79	1,79	**1,78**	7,40	7,37	**7,33**
	600	1,78	1,77	**1,75**	7,40	7,33	**7,27**
	1800	1,78	1,76	**1,75**	7,37	7,27	**7,07**
$\|V\| = 400$	30	2,88	2,88	**2,86**	7,57	7,53	**7,47**
	120	2,87	2,86	**2,83**	7,47	7,47	**7,40**
	600	2,86	2,85	**2,83**	7,43	7,40	**7,33**
	1800	2,84	2,83	**2,82**	7,40	7,33	**7,27**

Tabela 11 – Evolução dos resultados do Híbrido para as instâncias EU100, EU 200 e EU400 para o problema MITOT

	Tempo (s)	GRASP	VNS	Híbrido
$\|V\| = 100$	30	204,12	**204,03**	**204,03**
	120	203,37	**203,17**	**203,17**
	600	202,48	**201,89**	**201,89**
	1800	201,84	201,60	**201,35**
$\|V\| = 200$	30	217,57	**216,51**	**216,51**
	120	216,13	**215,38**	**215,38**
	600	215,88	215,02	**214,91**
	1800	215,21	214,74	**214,69**
$\|V\| = 400$	30	232,81	**231,72**	**231,72**
	120	229,36	228,71	**228,23**
	600	228,45	227,97	**227,05**
	1800	228,13	227,63	**227,05**

6.2.5 Comportamento das metaheurísticas GRASP, VNS e Híbrido para o problema MITOTP

Neste experimento, cada metaheurística foi executada durante um tempo máximo de 1800 segundos. A tabela 12 apresenta os resultados médios das 15 execuções para cada metaheurística para o problema MITOTP para as instâncias EU100, EU200 e EU400, sendo EU100 uma instância euclidiana com 100 nós, EU200 uma instância euclidiana com 200 nós e EU400 uma instância euclidiana com 400 nós. Na tabela, são apresentados os valores de potência e de interferência total e os valores em negrito destacam o melhor resultado dentro de cada tempo estabelecido.

Tabela 12 – Evolução dos resultados do Híbrido para as instâncias EU100, EU 200 e EU400 para o problema MITOTP

		Potência			Interferência Total		
	Tempo (s)	GRASP	VNS	Híbrido	GRASP	VNS	Híbrido
$\|V\| = 100$	30	1,47	**1,45**	**1,45**	203,04	**202,73**	**202,73**
	120	1,45	**1,44**	**1,44**	202,25	**202,16**	**202,16**
	600	1,44	1,43	**1,42**	201,57	200,75	**200,23**
	1800	1,43	**1,42**	**1,42**	200,67	200,26	**200,03**
$\|V\| = 200$	30	1,82	1,81	**1,80**	214,29	**213,45**	**213,45**
	120	1,80	1,80	**1,79**	212,89	**212,12**	**212,12**
	600	1,80	1,78	**1,77**	212,65	212,07	**211,78**
	1800	1,79	1,78	**1,77**	212,38	211,64	**211,56**
$\|V\| = 400$	30	2,91	**2,87**	**2,87**	231,25	230,82	**230,79**
	120	2,89	2,88	**2,87**	230,53	229,72	**229,44**
	600	2,88	2,86	**2,85**	229,37	228,93	**228,51**
	1800	2,87	2,85	**2,84**	229,17	228,88	**228,46**

6.2.6 Comparações das metaheurísticas com resultados ótimos

Neste segundo experimento, a principal métrica utilizada para avaliação do desempenho das metaheurísticas para $|V| \in \{10, 15, 20, 25, 30, 35, 40, 45, 50\}$ é encontrar os valores de solução ótima para os problemas MPT, MIMAX, MITOT, MIMAXP e MITOTP sobre restrição de biconexidade para instâncias euclidianas EU.

Os resultados obtidos pelas metaheurísticas para $|V| \in \{10, 15, 20, 25, 30, 35, 40, 45, 50\}$ foram comparados entre eles e com os valores obtidos pelas formulações matemáticas resolvidas pelo CPLEX. Os resultados dos experimentos realizados com o *solver* CPLEX foram obtidos de (MORAES, 2009).

Tabela 13 – Número de instâncias resolvidas otimamente em seis horas.

	MPT	MIMAX	MIMAXP	MITOT	MITOTP
$\|V\|$	Qtde	Qtde	Qtde	Qtde	Qtde
10	15	15	15	15	15
15	15	15	15	15	15
20	15	15	15	15	15
25	15	15	15	15	15
30	15	14	13	14	14
35	14	9	7	7	12
40	12	5	4	5	4
45	8	2	2	1	1
50	4	2	3	0	0

Na Tabela 13, apresenta-se o número de instâncias solucionadas de forma exata por cada formulação e para cada problema: MPT, MIMAX, MITOT, MIMAXP e MITOTP. Vê-se que o problema MPT obtém maior quantidade de soluções ótimas dentro do tempo de seis horas para todos os tamanhos de instâncias. Células que estão com valor zero mostram que o CPLEX não encontrou nenhuma solução no tempo limite.

A Tabela 14 mostra, para cada tamanho de V e para cada formulação, o tempo médio (em segundos) em que soluções ótimas foram encontradas por todas as formulações dentro de seis horas de processamento. Instâncias que a solução ótima não foi encontrada por uma dada formulação dentro de seis horas foram descartados nos resultados médios apresentados. A primeira coluna da Tabela 14 exibe entre parênteses o número de instâncias usadas para calcular as médias. As células em branco correspondem a formulações com menor número de instâncias resolvidas para a otimização, conforme mostrado na Tabela 13. Pode-se observar na Tabela 14 que a adição de restrições de interferência faz aumentar o tempo computacional gasto para resolver instâncias. Pela tabela, pode-se observar que, com o aumento do tamanho do problema, as formulações MIMAXP e MITOTP tornam-se mais rápidas do que os modelos MIMAX e MITOT.

Tabela 14 – Tempo médio em segundos para encontrar a solução ótima em todas as formulações.

$\|V\|$	MPT	MIMAX	MIMAXP	MITOT	MITOTP
10 (15)	0,32	0,25	0,79	0,96	1,59
15 (15)	1,89	5,06	5,34	7,27	9,63
20 (15)	17,02	70,76	39,12	75,27	28,98
25 (15)	109,51	1141,99	732,92	600,66	342,73
30 (15)	329,90	2471,91	2988,87	2735,39	2914,51
35 (14)	332,16	5493,89	2012,23	6035,40	2649,74
40 (12)	7134,33	–	–	–	–
45 (8)	8928,67	–	–	–	–
50 (4)	13790,09	–	–	–	–

6.2.6.1 Minimização de potência (MPT)

Neste experimento, a métrica analisada é a minimização da potência das RSSF para $|V|$ entre 10 à 50 nós. Cada instância foi executada uma vez, sendo aplicado como critério de parada o tempo máximo de 30 segundos ou o valor da solução ótima, o que ocorrer primeiro.

A Tabela 15 apresenta o valor médio de potência total obtida do resultado de 15 instâncias para cada tamanho de V resolvida pelo CPLEX e pelas metaheurísticas GRASP, VNS e Híbrida (GRASP-VNS). Os melhores resultados estão marcados em negrito (valores ótimos obtidos pelo CPLEX). Para o problema de minimização de potência, a Tabela 15 mostra que a metaheurística Híbrida foi a única que encontrou todas as soluções ótimas conhecidas em menos de 30 segundos. Quando encontrados, todos os resultados ótimos da Tabela 15 foram alcançados em menos de 1,68 segundos.

6.2.6.2 Minimização de interferência máxima (MIMAX)

Neste outro experimento, a métrica analisada é a minimização da interferência máxima para redes com $|V|$ entre 10 à 50 nós. Os resultados detalhados são apresentados na Tabela 16 para as metaheurísticas GRASP, VNS e GRASP-VNS. Para cada instância, indicamos, também, o valor da solução ótima encontrada pela formulação matemática

Tabela 15 – Média dos resultados obtidos para minimização de potência (MPT) com $|V|$ entre 10 e 50 nós

$\|V\|$	GRASP	VNS	Híbrido	Ótimo
10 (15)	**1,66**	**1,66**	**1,66**	**1,66**
15 (15)	**1,44**	**1,44**	**1,44**	**1,44**
20 (15)	**1,45**	**1,45**	**1,45**	**1,45**
25 (15)	**1,36**	**1,36**	**1,36**	**1,36**
30 (15)	**1,32**	**1,32**	**1,32**	**1,32**
35 (14)	**1,26**	**1,26**	**1,26**	**1,26**
40 (12)	**1,25**	**1,25**	**1,25**	**1,25**
45 (8)	**1,23**	**1,23**	**1,23**	**1,23**
50 (4)	1,29	1,25	**1,21**	**1,21**

resolvida pelo CPLEX. Estes valores estão marcados em negrito na tabela. Os valores de interferência máxima apresentados na Tabela 16 são valores médios obtidos do resultado de uma execução de cada uma das 15 instâncias para cada tamanho de V. Os critérios de parada utilizados foram os mesmos usados para minimização da potência.

Também é mostrada a diferença percentual entre os valores obtidos pelas metaheurísticas e a solução ótima. Para o problema de minimização de interferência máxima, a Tabela 16 mostra que a diferença percentual dos resultados da metaheurística Híbrida para a solução ótima para $|V|$ de 10 a 50 nós é 0,00%, ou seja, novamente, a metaheurística Híbrida foi a única que encontrou todas as soluções ótimas conhecidas em menos de 30 segundos. Comparando os resultados da solução ótima com o GRASP, a maior diferença percentual é de 9,40%. A maior diferença percentual comparando VNS com a solução ótima é de 6,60%.

Tabela 16 – Média dos resultados obtidos para minimização da interferência máxima (MIMAX) com $|V|$ entre 10 e 50 nós

	GRASP		VNS		Híbrido		
|V|	Média	Dif. (%)	Média	Dif. (%)	Média	Dif. (%)	Ótimo
10 (15)	**5,07**	(0,00)	**5,07**	(0,00)	**5,07**	(0,00)	**5,07**
15 (15)	**4,93**	(0,00)	**4,93**	(0,00)	**4,93**	(0.00)	**4,93**
20 (15)	**5,07**	(0,00)	**5,07**	(0,00)	**5,07**	(0,00)	**5,07**
25 (15)	**5,33**	(0,00)	**5,33**	(0,00)	**5,33**	(0,00)	**5,33**
30 (14)	**5,00**	(0,00)	**5,00**	(0,00)	**5,00**	(0,00)	**5,00**
35 (9)	**5,09**	(0,00)	**5,09**	(0,00)	**5,09**	(0,00)	**5,09**
40 (5)	**5,50**	(0,00)	**5,50**	(0,00)	**5,50**	(0,00)	**5,50**
45 (2)	5,47	(9,40)	5,33	(6,60)	**5,00**	(0,00)	**5,00**
50 (2)	5,47	(9,40)	5,27	(5,40)	**5,00**	(0,00)	**5,00**

6.2.6.3 Minimização de interferência total (MITOT)

Neste terceiro experimento, a métrica analisada é a minimização da interferência total para redes com $|V|$ entre 10 à 50 nós. Os resultados detalhados são apresentados na Tabela 17 para as metaheurísticas GRASP, VNS e Híbrida. Para cada instância, indicamos, também, a solução ótima encontrada pela formulação matemática resolvida pelo CPLEX. Estes resultados estão marcados em negrito. Para tamanhos de $|V|$ na qual a formulação exata não pode solucionar as quinze instâncias, é marcado em negrito o melhor resultado, que é obtido por uma metaheurística. Além disso, é mostrada a diferença percentual entre este melhor resultado (marcado em negrito) e o resultado obtido pelas outras metaheurísticas. Os valores de interferência total apresentados na Tabela 17 são valores médios obtidos do resultado de uma execução de cada uma das 15 instâncias para cada tamanho de V. Os critérios de parada utilizados foram os mesmos usados para minimização da potência.

Para o problema de minimização de interferência total, a Tabela 17 mostra que a diferença percentual dos resultados do Híbrido para a solução ótima para $|V|$ de 10 a 45 nós é 0.00%. O GRASP e o VNS obtém os valores de solução ótima para $|V|$ de 10 a 40 nós. Para $|V| = 45$, o resultado obtido pelo VNS é 5,60% distante do ótimo e o resultado obtido pelo GRASP é 7,50% distante do ótimo.

Para $|V|$ igual a 50 nós, a formulação exata não consegue resolver nenhuma instância (ver Tabela 13). Então, o algoritmo que obtém os melhores resultados é a metaheurística Híbrida. Neste caso, a diferença percentual do GRASP para o Híbrido é de 5,07%. A diferença percentual do VNS para o Híbrido é de 3,19%.

Tabela 17 – Média dos resultados obtidos para minimização da interferência total (MITOT) com V entre 10 e 50 nós

	GRASP		VNS		Híbrido		
|V|	Média	Dif. (%)	Média	Dif. (%)	Média	Dif. (%)	Ótimo
10 (15)	**33,00**	(0,00)	**33,00**	(0,00)	**33,00**	(0,00)	**33,00**
15 (15)	**48,60**	(0,00)	**48,60**	(0,00)	**48,60**	(0,00)	**48,60**
20 (15)	**65,60**	(0,00)	**65,60**	(0,00)	**65,60**	(0,00)	**65,60**
25 (15)	**80,87**	(0,00)	**80,87**	(0,00)	**80,87**	(0,00)	**80,87**
30 (14)	**97,53**	(0,00)	**97,53**	(0,00)	**97,53**	(0,00)	**97,53**
35 (7)	**204,18**	(0,00)	**204,18**	(0,00)	**204,18**	(0,00)	**204,18**
40 (5)	**118,25**	(0,00)	**118,25**	(0,00)	**118,25**	(0,00)	**118,25**
45 (1)	134,38	(7,50)	132,00	(5,60)	**125,00**	(0,00)	**125,00**
50 (0)	191,45	(5,07)	188,33	(3,19)	**182,50**	(0,00)	-

6.2.6.4 Minimização de interferência máxima e potência (MIMAXP)

Neste quarto experimento, a métrica analisada é a minimização da interferência máxima e também a minimização da potência para redes com $|V|$ entre 10 à 50 nós. Os resultados detalhados para minimização de potência e minimização de interferência máxima são apresentados nas Tabelas 18 e 19 para as metaheurísticas GRASP, VNS e Híbrida. Para cada instância, indicamos também a solução ótima encontrada pela formulação matemática resolvida pelo CPLEX. Os valores apresentados nas tabelas são valores médios obtidos a partir da execução de cada uma das 15 instâncias para cada tamanho de V. Os critérios de parada utilizados foram os mesmos para a minimização da potência.

Nas Tabelas 18 e 19, enquanto a metaheurística Híbrida encontra todos os valores ótimos conhecidos dentro do limite de 30 segundos, os demais algoritmos só encontram a solução ótima para as instâncias até o tamanho $|V| = 30$. Na Tabela 18, comparando os resultados da solução ótima com o GRASP, a maior diferença percentual é de 5,93%, para $|V| = 45$. A maior diferença percentual do VNS é de 3,39%, também para $|V| = 45$.

Tabela 18 – Média dos resultados obtidos para minimização da interferência máxima e potência (MIMAXP) com $|V|$ entre 10 e 50 nós - Potência

	GRASP		VNS		Híbrido		
|V|	Média	Dif. (%)	Média	Dif. (%)	Média	Dif. (%)	Ótimo
10 (15)	**1,69**	(0,00)	**1,69**	(0,00)	**1,69**	(0,00)	**1,69**
15 (15)	**1,48**	(0,00)	**1,48**	(0,00)	**1,48**	(0,00)	**1,48**
20 (15)	**1,49**	(0,00)	**1,49**	(0,00)	**1,49**	(0,00)	**1,49**
25 (15)	**1,40**	(0,00)	**1,40**	(0,00)	**1,40**	(0,00)	**1,40**
30 (13)	**1,35**	(0,00)	**1,35**	(0,00)	**1,35**	(0,00)	**1,35**
35 (7)	1,36	(2,25)	1,34	(0,75)	**1,33**	(0,00)	**1,33**
40 (4)	1,18	(2,60)	1,16	(0,87)	**1,15**	(0,00)	**1,15**
45 (2)	1,25	(5,93)	1,22	(3,39)	**1,18**	(0,00)	**1,18**
50 (3)	1,21	(4,31)	1,18	(1,72)	**1,16**	(0,00)	**1,16**

Tabela 19 – Média dos resultados obtidos para minimização da interferência máxima e potência (MIMAXP) com $|V|$ entre 10 e 50 nós - Interferência máxima

	GRASP		VNS		Híbrido		
|V|	Média	Dif. (%)	Média	Dif. (%)	Média	Dif. (%)	Ótimo
10 (15)	**5,07**	(0,00)	**5,07**	(0,00)	**5,07**	(0,00)	**5,07**
15 (15)	**4,93**	(0,00)	**4,93**	(0,00)	**4,93**	(0,00)	**4,93**
20 (15)	**5,07**	(0.00)	**5,07**	(0,00)	**5,07**	(0,00)	**5,07**
25 (15)	**5,33**	(0,00)	**5,33**	(0,00)	**5,33**	(0,00)	**5,33**
30 (13)	**5,00**	(0,00)	**5,00**	(0,00)	**5,00**	(0,00)	**5,00**
35 (7)	5,27	(5,40)	5,27	(5,40)	**5,00**	(0,00)	**5,00**
40 (4)	5,67	(3,10)	**5,50**	(0,00)	**5,50**	(0,00)	**5,50**
45 (2)	5,33	(6,60)	5,27	(5,40)	**5,00**	(0,00)	**5,00**
50 (3)	5,47	(9,40)	5,27	(5,40)	**5,00**	(0,00)	**5,00**

Na Figura 21, tem-se dois gráficos de barras. O primeiro representa o valor de potência médio encontrado por cada metaheurística para cada tamanho de V, conforme apresentado na Tabela 18, enquanto o segundo gráfico, representa o valor de interferência máxima médio para cada tamanho de V, conforme visto na Tabela 19. Destes gráficos, pode-se concluir que a metaheurística Híbrida encontra melhores resultados, comparados com o GRASP e VNS. Além disso, percebe-se que os valores de potência diminuem a medida que $|V|$ aumenta, pois as instâncias testadas são euclidianas EU, na qual é fixado o tamanho da área onde os nós são distribuídos. Então, quanto maior o número de nós, maior a quantidade de nós próximos um do outro e menor o valor de potência.

Na Tabela 20, são apresentados os tempos de execução médio de cada metaheurística, em segundos, para cada tamanho de V. Como esperado, a metaheurística Híbrida é mais robusta pois mantém-se achando as soluções ótimas para $|V|$ maior que 35 em pouco tempo, comparando-a com o GRASP e VNS.

6.2.6.5 Minimização de interferência total e potência (MITOTP)

Neste último experimento, a métrica analisada é a minimização da interferência total e também a minimização da potência para redes com $|V|$ entre 10 à 50 nós. Os

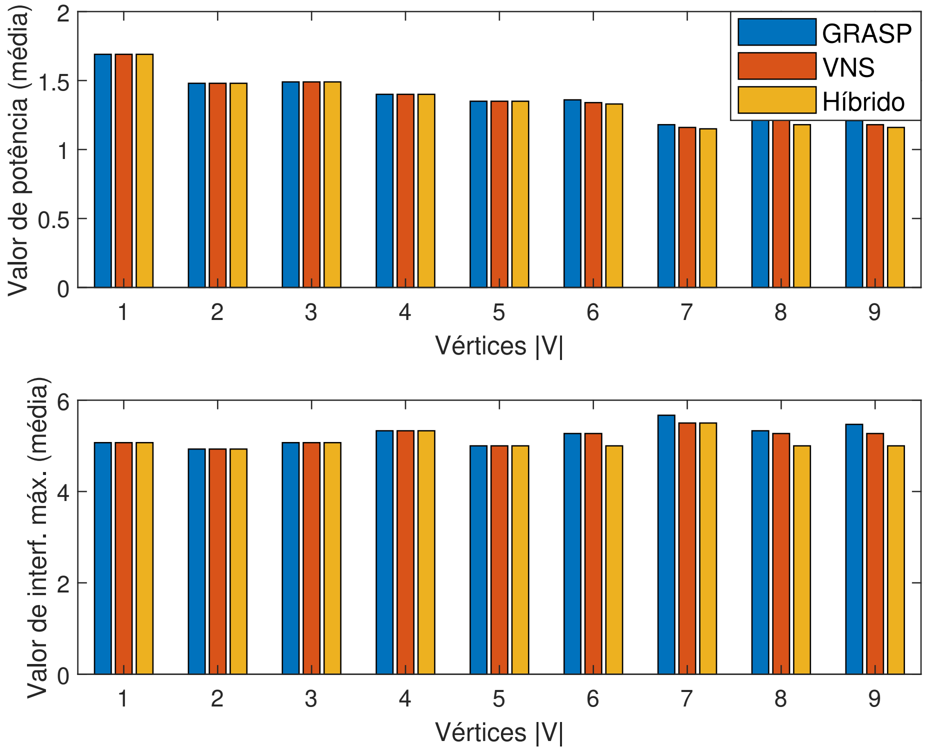


Figura 21 – Evolução dos valores de potência e interferência máxima para as metaheurísticas GRASP, VNS e Híbrida para MIMAXP - Instância EU

Tabela 20 – Tempo médio (em segundos) - MIMAXP

$\|V\|$	GRASP	VNS	Híbrido
10	1,09	1,57	1,62
15	1,33	1,84	1,98
20	1,35	1,63	1,77
25	2,29	2,58	2,64
30	2,30	2,84	2,86
35	30	30	3.02
40	30	30	3.14
45	30	30	3.75
50	30	30	3.82

resultados detalhados para minimização de potência e minimização de interferência total são apresentados nas Tabelas 21 e 22 para as metaheurísticas GRASP, VNS e GRASP-VNS, respectivamente. Para cada instância, indicamos, também, a solução ótima encontrada pela formulação matemática resolvida pelo CPLEX. Como já afirmado anteriormente, os valores apresentados nas tabelas são valores médios obtidos do resultado de uma execução de cada uma das 15 instâncias para cada tamanho de V. Os critérios de parada utilizados foram os mesmos da minimização da potência.

Nas Tabelas 21 e 22, vê-se que a metaheurística Híbrida encontra os mesmos valores da solução ótima para todas as instâncias na qual o ótimo é conhecido. Analisando a Tabela 21, vê-se que o GRASP não encontra a solução ótima no tempo fornecido para $|V| \geq 25$ nós. O VNS não encontra a solução ótima no tempo fornecido para $|V| \geq 35$

nós. Nestes casos, comparando os resultados da solução ótima com o GRASP, a maior diferença percentual é de 6,67%, para $|V| = 40$. A maior diferença percentual do VNS é de 4,17%, para $|V| = 40$.

Tabela 21 – Média dos resultados obtidos para minimização da interferência total e potência (MITOTP) com $|V|$ entre 10 e 50 nós - Potência

	GRASP		VNS		Híbrido		
|V|	Média	Dif. (%)	Média	Dif. (%)	Média	Dif. (%)	Ótimo
10 (15)	**1,73**	(0,00)	**1,73**	(0,00)	**1,73**	(0,00)	**1,73**
15 (15)	**1,50**	(0,00)	**1,50**	(0,00)	**1,50**	(0,00)	**1,50**
20 (15)	**1,53**	(0,00)	**1,53**	(0,00)	**1,53**	(0,00)	**1,53**
25 (15)	1,48	(2,77)	**1,44**	(0,00)	**1,44**	(0,00)	**1,44**
30 (14)	1,37	(2,23)	**1,34**	(0,00)	**1,34**	(0,00)	**1,34**
35 (12)	1,34	(3,87)	1,33	(3,10)	**1,29**	(0,00)	**1,29**
40 (4)	1,28	(6,67)	1,25	(4,17)	**1,20**	(0,00)	**1,20**
45 (1)	1,18	(5,36)	1,16	(3,57)	**1,12**	(0,00)	**1,12**
50 (0)	1,10	(2,80)	1,08	(0,93)	**1,07**	(0,00)	**-**

Tabela 22 – Média dos resultados obtidos para minimização da interferência total e potência (MITOTP) com $|V|$ entre 10 e 50 nós - Interferência Total

	GRASP		VNS		Híbrido		
|V|	Média	Dif. (%)	Média	Dif. (%)	Média	Dif. (%)	Ótimo
10 (15)	**33,00**	(0,00)	**33,00**	(0,00)	**33,00**	(0,00)	**33,00**
15 (15)	**48,60**	(0,00)	**48,60**	(0,00)	**48,60**	(0,00)	**48,60**
20 (15)	**65,60**	(0,00)	**65,60**	(0,00)	**65,60**	(0,00)	**65,60**
25 (15)	87,46	(8,15)	**80,87**	(0,00)	**80,87**	(0,00)	**80,87**
30 (14)	102,53	(6,58)	**96,20**	(0,00)	**96,20**	(0,00)	**96,20**
35 (12)	113,57	(4,02)	111,84	(2,44)	**109,18**	(0,00)	**109,18**
40 (4)	127,38	(4,41)	125,28	(2,69)	**122,00**	(0,00)	**122,00**
45 (1)	128,35	(2,68)	125,78	(0,62)	**125,00**	(0,00)	**125,00**
50 (0)	153,56	(1,84)	152,56	(1,18)	**150,78**	(0,00)	**-**

Para $|V| = 50$ nós, como a formulação exata não consegue resolver todas as quinze instâncias, o algoritmo que obtém os melhores resultados é a metaheurística Híbrida. Neste caso, comparando os resultados do Híbrido com o GRASP, a diferença percentual é de 2,80%. A diferença percentual do VNS é de 0,93%.

Na Tabela 22, comparando os resultados da solução ótima com o GRASP, a maior diferença percentual é de 6,58%. A maior diferença percentual do VNS é de 2,69%. Para $|V| = 50$ nós, comparando os resultados do Híbrido com o GRASP, a diferença percentual é de 1,84%. A diferença percentual do VNS é de 1,18%.

Na Figura 22, o primeiro gráfico representa o valor de potência médio encontrado por cada metaheurística para cada tamanho de $|V|$, conforme apresentado na Tabela 21. No segundo gráfico, vê-se representado o valor de interferência total médio para cada tamanho de $|V|$, conforme visto na Tabela 22.

Destes gráficos, pode-se concluir que a metaheurística Híbrida encontra os melhores resultados, comparados com o GRASP e VNS. Conforme discutido anteriormente, os valores de potência diminuem a medida que $|V|$ aumenta.

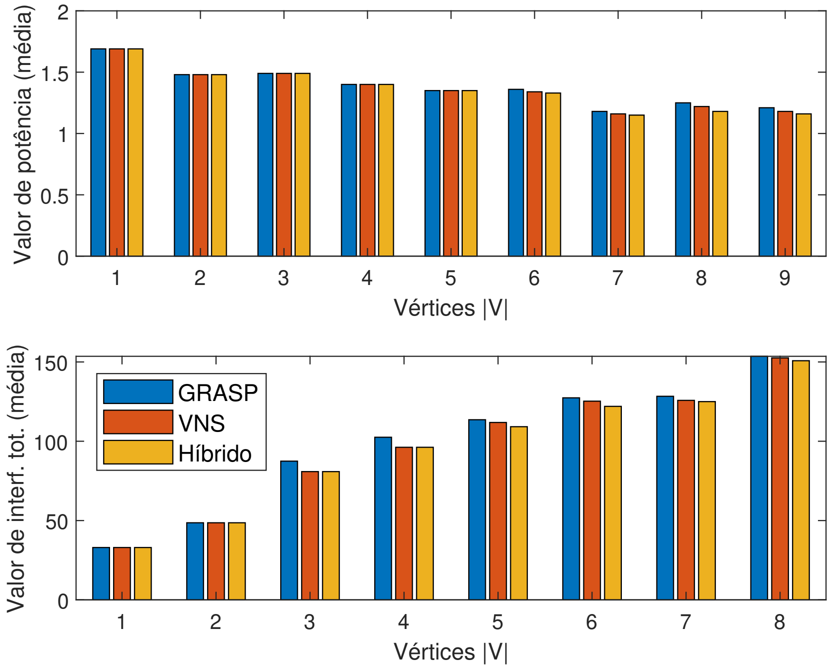


Figura 22 – Evolução dos valores de potência e interferência total para as metaheurísticas GRASP, VNS e Híbrido para MITOTP - Instância EU

Os resultados mostrados nas tabelas acima foram obtidos para a instância EU, pois são as instâncias com as soluções ótimas conhecidas. Através dos resultados obtidos, pode-se concluir que, para o modelo MPT, as metaheurísticas se aproximam mais dos valores de soluções ótimas, pois não há restrições de interferência. Quando minimiza-se a interferência (problemas MIMAX e MITOT), a metaheurística Híbrida já se mostra mais eficiente para instâncias maiores do que 45 nós na apresentação de resultados. Quando minimiza-se ambos (problemas MIMAXP e MITOTP), a metaheurística Híbrida já se mostra mais eficiente para instâncias maiores do que 25 nós.

6.2.7 Resultados obtidos com o Algoritmo Híbrido para instâncias realísticas

Nesta seção, aplica-se a metaheurística Híbrida em problemas de tamanho realístico para os tipos de instâncias EU, DE e RD. Os experimentos para instâncias realísticas foram realizados com o algoritmo Híbrido, porque se mostrou mais eficiente na obtenção dos valores ótimos conhecidos para as instâncias de pequeno porte, como mostrado na Seção 6.2.6.

6.2.7.1 Comparação entre instâncias

Os resultados obtidos para minimização de potência (MPT), minimização de interferência máxima (MIMAX), minimização de interferência total (MITOT), minimização de potência e interferência máxima (MIMAXP) e minimização de potência e interferência total (MITOTP) são mostrados nas tabelas a seguir. O critério de parada utilizado foi o tempo de processamento de 1800s. Comparando os resultados mostrados na Tabela 23, vê-se que os valores de potência do problema MPT são menores para todos os tamanhos de instância comparado com os outros problemas. Isso também vale para as instâncias DE e RD, como pode ser visto nas Tabelas 24 e 25, respectivamente.

Tabela 23 – Comparação dos valores de potência para instância EU

	MITOTP		MITOT		MIMAXP		MIMAX		MPT
|V|	Média	Dif.	Média	Dif.	Média	Dif.	Média	Dif.	Média
25	1,44	(5,88)	1,45	(6,61)	1,40	(2,94)	2,05	(50,73)	**1,36**
50	1,35	(6,30)	1,38	(8,66)	1,32	(3,94)	1,78	(40,16)	**1,27**
100	1,42	(5,18)	1,47	(8,88)	1,39	(2,96)	1,95	(44,44)	**1,35**
200	1,77	(4,73)	1,80	(6,50)	1,75	(3,55)	2,17	(28,40)	**1,69**
400	2,84	(1,06)	2,87	(2,13)	2,82	(0,35)	3,02	(7,47)	**2,81**
800	5,06	(0,40)	5,08	(0,79)	5,05	(0,20)	5,39	(6,94)	**5,04**

Tabela 24 – Comparação dos valores de potência para instância DE

	MITOTP		MITOT		MIMAXP		MIMAX		MPT
|V|	Média	Dif.	Média	Dif.	Média	Dif.	Média	Dif.	Média
25	5,55	(1,65)	5,59	(2,38)	5,51	(0,91)	5,68	(4,03)	**5,46**
50	10,29	(0,78)	10,35	(1,37)	10,24	(0,29)	10,47	(2,55)	**10,21**
100	19,52	(0,51)	19,58	(0,82)	19,48	(0,31)	19,76	(1,75)	**19,42**
200	37,64	(0,32)	37,71	(0,50)	37,58	(0,16)	37,94	(1,12)	**37,52**
400	74,23	(0,15)	74,33	(0,28)	74,17	(0,07)	75,12	(1,35)	**74,12**
800	109,64	(0,93)	112,89	(3,92)	109,25	(0,57)	113,23	(4,23)	**108,63**

Tabela 25 – Comparação dos valores de potência para instância RD

	MITOTP		MITOT		MIMAXP		MIMAX		MPT
|V|	Média	Dif.	Média	Dif.	Média	Dif.	Média	Dif.	Média
25	5,65	(2,35)	5,72	(3,62)	5,57	(0,90)	5,91	(7,06)	**5,52**
50	8,55	(1,18)	8,63	(2,13)	8,47	(0,23)	8,87	(4,97)	**8,45**
100	11,89	(1,36)	11,97	(2,05)	11,80	(0,60)	12,27	(4,60)	**11,73**
200	17,77	(1,31)	17,93	(2,22)	17,64	(0,57)	18,32	(4,45)	**17,54**
400	25,27	(0,55)	25,43	(1,19)	25,19	(0,23)	25,95	(3,26)	**25,13**
800	41,63	(5,47)	41,89	(6,13)	40,82	(3,42)	42,64	(8,03)	**39,47**

Uma análise dos resultados apresentados na Tabela 26 permite verificar que os problemas MIMAX e MIMAXP apresentam menores interferências para todos os tamanhos de instância. Isso também vale para as instâncias DE e RD, como pode ser visto nas Tabelas 27 e 28, respectivamente. Os maiores valores de interferência máxima são encontrados no problema MPT.

Analisando também os resultados apresentados na Tabela 29, verifica-se que MITOT e MITOTP proporcionam as menores interferências totais para todos os tamanhos de instância. Isso também vale para as instâncias DE e RD, como pode ser visto nas Tabelas 30 e 31, respectivamente.

A Tabela 32 mostra os resultados para MIMAX e também para MIMAXP para todos os tipos de instâncias. No problema MIMAX, os valores médios de potência são

Tabela 26 – Comparação dos valores de interferência máxima para instância EU

	MPT		MITOT		MITOTP		MIMAXP		MIMAX
|V|	Média	Dif.	Média	Dif.	Média	Dif.	Média	Dif.	Média
25	5,67	(6,38)	5,80	(8,82)	5,73	(7,50)	5,33	(0,00)	**5,33**
50	6,25	(25,00)	6,07	(21,40)	6,00	(20,00)	5,00	(0,00)	**5,00**
100	7,27	(14,85)	6,47	(2,21)	6,47	(2,21)	6,33	(0,00)	**6,33**
200	7,33	(3,67)	7,27	(2,83)	7,27	(2,83)	7,07	(0,00)	**7,07**
400	7,83	(6,82)	7,33	(0,82)	7,33	(0,82)	7,27	(0,00)	**7,27**
800	8,00	(2,17)	7,93	(1,27)	7,93	(1,27)	7,83	(0,00)	**7,83**

Tabela 27 – Comparação dos valores de interferência máxima para instância DE

	MPT		MITOT		MITOTP		MIMAXP		MIMAX
|V|	Média	Dif.	Média	Dif.	Média	Dif.	Média	Dif.	Média
25	6,00	(9,69)	5,80	(6,03)	5,83	(6,58)	5,67	(3,65)	**5,47**
50	7,33	(5,77)	7,27	(4,90)	7,00	(1,01)	6,93	(0,00)	**6,93**
100	7,47	(2,75)	7,33	(0,82)	7,33	(0,82)	7,27	(0,00)	**7,27**
200	8,67	(8,37)	8,53	(6,62)	8,27	(3,37)	8,00	(0,00)	**8,00**
400	10,53	(5,30)	10,27	(2,70)	10,27	(2,70)	10,00	(0,00)	**10,00**
800	10,83	(5,45)	10,47	(1,95)	10,47	(1,95)	10,33	(0,58)	**10,27**

Tabela 28 – Comparação dos valores de interferência máxima para instância RD

	MPT		MITOT		MITOTP		MIMAX		MIMAXP
|V|	Média	Dif.	Média	Dif.	Média	Dif.	Média	Dif.	Média
25	5,67	(6,38)	5,67	(6,38)	5,53	(3,75)	5,47	(2,62)	**5,33**
50	6,27	(3,29)	6,33	(4,28)	6,27	(3,29)	6,07	(0,00)	**6,07**
100	7,27	(3,85)	7,27	(3,85)	7,27	(3,85)	7,00	(0,00)	**7,00**
200	7,47	(2,75)	7,47	(2,75)	7,47	(2,75)	7,33	(0,82)	**7,27**
400	7,83	(6,82)	7,67	(4,64)	7,67	(4,64)	7,47	(1,90)	**7,33**
800	8,27	(5,62)	8,00	(2,17)	8,00	(2,17)	7,83	(0,00)	**7,83**

maiores do que os valores obtidos na formulação MIMAXP, mas os valores de interferência máxima são semelhantes para ambos. No modelo MIMAXP, vê-se que há uma redução tanto dos valores médios de potência. Ou seja, é preferível resolver MIMAXP. Resultados semelhantes são obtidos quando se compara as formulações MITOT e MITOTP, conforme pode ser observado na Tabela 33.

As Figuras 23 e 24 representam comparações entre os problemas MIMAX e MIMAXP para as instâncias EU e DE. Os valores utilizados nos gráficos são os valores representados na Tabela 32. As tabelas apresentadas são uma amostra representativa deste experimento e os resultados para todas as instâncias para os cinco problemas podem ser conferidos no Apêndice A.

Tabela 29 – Comparação dos valores de interferência total para instância EU

	MPT		MIMAX		MIMAXP		MITOT		MITOTP
|V|	Média	Dif.	Média	Dif.	Média	Dif.	Média	Dif.	Média
25	83,00	(2,63)	112,67	(39,32)	83,53	(3,29)	80,87	(0,00)	**80,87**
50	153,75	(1,97)	223,50	(48,23)	152,30	(1,00)	182,50	(21,03)	**150,78**
100	207,36	(3,66)	214,39	(7,18)	213,58	(6,77)	201,35	(0,66)	**200,03**
200	216,35	(2,26)	223,64	(5,70)	221,82	(4,85)	214,69	(1,48)	**211,56**
400	232,68	(1,85)	239,35	(4,76)	233,68	(2,28)	**227,05**	(-0,62)	228,46
800	312,37	(1,82)	309,68	(0,94)	309,12	(0,76)	307,08	(0,09)	**306,78**

Tabela 30 – Comparação dos valores de interferência total para instância DE

	MPT		MIMAX		MIMAXP		MITOT		MITOTP
|V|	Média	Dif.	Média	Dif.	Média	Dif.	Média	Dif.	Média
25	115,25	(7,89)	110,52	(3,46)	109,75	(2,74)	108,75	(1,80)	**106,82**
50	205,59	(3,19)	202,75	(1,46)	203,12	(1,95)	200,62	(0,69)	**199,23**
100	236,78	(3,18)	233,34	(1,69)	230,34	(0,38)	231,13	(0,72)	**229,47**
200	388,29	(2,83)	382,28	(1,24)	381,94	(1,15)	379,89	(0,61)	**377,59**
400	565,67	(1,02)	563,86	(0,69)	563,52	(0,63)	562,04	(0,36)	**559,98**
800	702,89	(1,09)	698,42	(0,44)	698,07	(0,39)	697,63	(0,33)	**695,33**

Tabela 31 – Comparação dos valores de interferência total para instância RD

	MPT		MIMAX		MIMAXP		MITOT		MITOTP
|V|	Média	Dif.	Média	Dif.	Média	Dif.	Média	Dif.	Média
25	101,14	(4,64)	103,65	(7,24)	102,23	(5,77)	98,35	(1,76)	**96,65**
50	183,64	(1,04)	188,76	(3,86)	184,46	(1,50)	182,98	(0,68)	**181,74**
100	220,64	(2,47)	222,92	(3,52)	218,69	(1,56)	217,25	(0,90)	**215,32**
200	232,75	(2,28)	233,84	(2,76)	231,72	(1,83)	229,37	(0,79)	**227,56**
400	242,37	(0,66)	246,27	(2,28)	242,52	(0,72)	243,85	(1,27)	**240,78**
800	342,75	(1,71)	338,27	(0,38)	337,78	(0,23)	337,92	(0,27)	**336,98**

Tabela 32 – Comparação entre MIMAX e MIMAXP para todas as instâncias

	EU				DE				RD			
	MIMAX		MIMAXP		MIMAX		MIMAXP		MIMAX		MIMAXP	
|V|	Pot	IM	Pot	IM	Pot	IM	Pot	IM	Pot	IM	Pot	IM
25	2,05	5,33	1,40	5,33	5,68	5,47	5,51	5,67	5,91	5,47	5,57	5,33
50	178	5,00	1,32	5,00	10,47	6,93	10,24	6,93	8,87	6,07	8,47	6,07
100	1,95	6,33	1,39	6,33	19,76	7,27	19,48	7,27	12,27	7,00	11,80	7,00
200	2,17	7,07	1,75	7,07	37,94	8,00	37,58	8,00	18,32	7,33	17,64	7,27
400	3,02	7,27	2,82	7,27	75,12	10,00	74,17	10,00	25,95	7,47	25,19	7,33
800	5,39	7,83	5,05	7,83	113,23	10,27	109,25	10,33	42,64	7,83	40,82	7,83

Tabela 33 – Comparação entre MITOT e MITOTP para todas as instâncias

	EU				DE				RD			
	MITOT		MITOTP		MITOT		MITOTP		MITOT		MITOTP	
|V|	Pot	IT	Pot	IT	Pot	IT	Pot	IT	Pot	IT	Pot	IT
25	1,45	80,87	1,44	80,87	5,59	108,75	5,55	106,82	5,72	98,35	5,65	96,65
50	1,38	182,50	1,35	150,78	10,35	200,62	10,29	199,23	8,63	182,98	8,55	181,74
100	1,47	201,35	1,42	200,03	19,58	231,13	19,52	229,47	11,97	217,25	11,89	215,32
200	1,80	214,69	1,77	211,56	37,71	379,89	37,64	377,47	17,93	229,37	17,77	227,56
400	2,87	227,05	2,84	228,46	74,33	562,04	74,23	559,98	25,43	243,85	25,27	240,78
800	5,08	307,08	5,08	306,78	112,89	697,63	109,64	695,33	41,89	337,92	41,63	336,98

As Figuras 25, 26, 27 e 28 comparam as metaheurísticas GRASP, VNS e Híbrida para as instâncias $|V| = 25$ e $|V| = 200$ usando a metodologia proposto por Aiex et al. (AIEX; RESENDE; RIBEIRO, 2007). Nas Figuras 25 e 26 vê-se a probabilidade acumulada de encontrar a solução ótima em função do tempo entre os algoritmos GRASP, VNS e Híbrido para a instância $|V| = 25$ para os problemas MITOT e MITOTP e nas Figuras 27 e 28 vê-se a probabilidade acumulada para a instância de $|V| = 200$ para os problemas MITOT e MITOTP. Os gráficos para os problemas MPT, MIMAX e MIMAXP são apresentados no Apêndice C.

Para traçar a distribuição empírica de cada algoritmo, foi associada uma probabili-

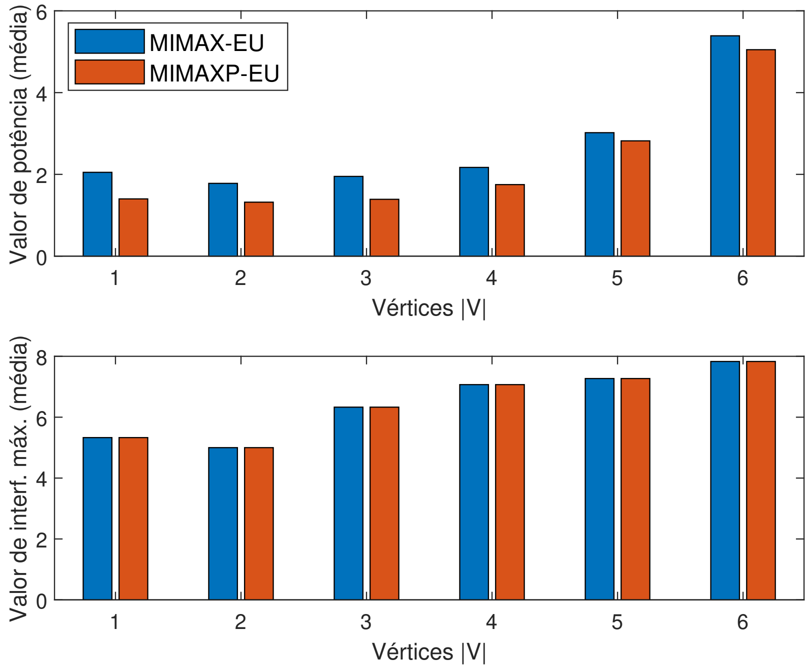


Figura 23 – Comparação entre MIMAX e MIMAXP - Instância EU

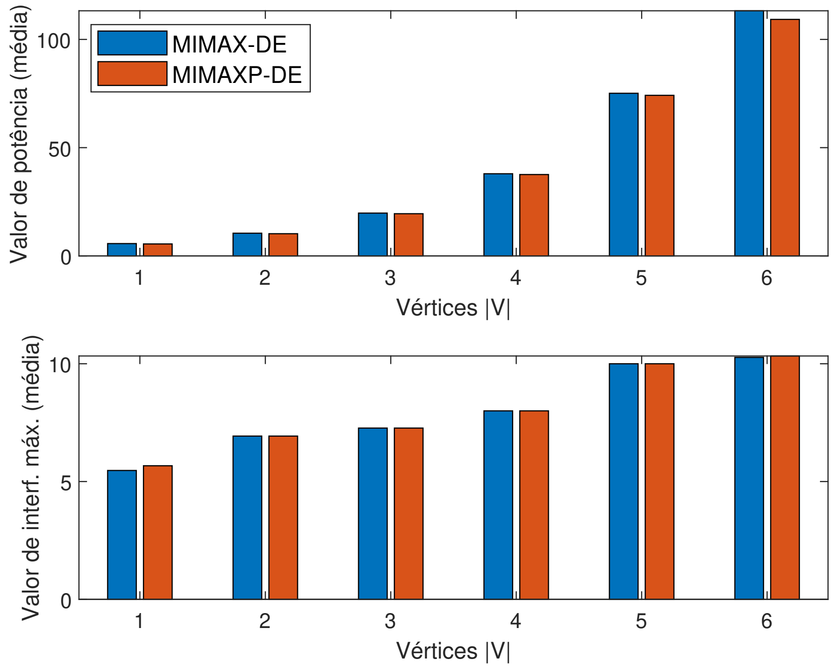


Figura 24 – Comparação entre MIMAX e MIMAXP - Instância DE

dade, como mostrada abaixo:

$$p_i = \frac{(i - \frac{1}{2})}{200} \tag{6.3}$$

com o i-ésimo menor tempo de execução t_i e traçou-se os pontos $z_i = (t_i, p_i)$ para $i = 1, ..., 200$. Duzentas simulações independentes foram executadas para cada algoritmo e para cada instância.

Foi utilizada a instância $|V| = 25$, pois são conhecidos os valores ótimos. Para $|V| = 25$, foi escolhida a instância 25AA. Também foi utilizada a instância $|V| = 200$ pois é considerada uma instância de tamanho realístico. Para $|V| = 200$, foi escolhida a instância 200AA. As instâncias 25AA e 200AA representam o padrão de comportamento e por isso foram utilizadas.

Para a instância $|V| = 25$, a execução foi encerrada quando foi obtido valor ótimo da função objetivo. As Figuras 25 e 26 mostram que, para alvos relativamente fáceis ($V = 25$), o algoritmo GRASP é mais eficiente que os outros porque o tamanho de sua vizinhança é o menor, levando a um procedimento de busca local mais rápido.

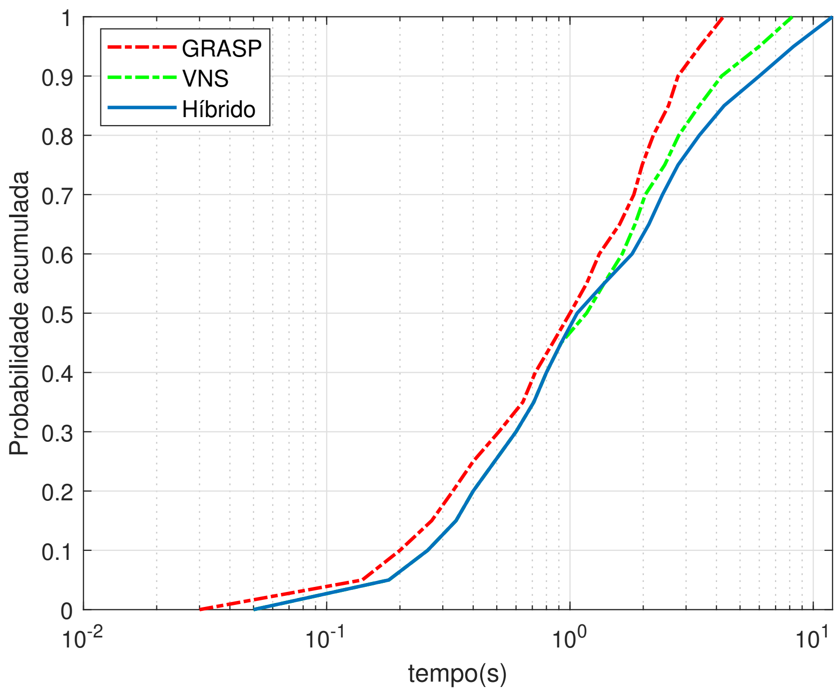


Figura 25 – Gráfico de probabilidade acumulada para EU25 - MITOT

Para a instância $|V| = 200$, a execução foi encerrada quando uma solução com valor igual a um alvo foi encontrado. Esse alvo foi escolhido baseado no valor da função objetivo do algoritmo que foi mais lento para $t = 600s$. As Figuras 27 e 28 mostram que o Híbrido se torna o algoritmo mais rápido quando valores de solução alvo mais difíceis são buscados. Essa se torna a principal desvantagem do GRASP, pois a probabilidade dele encontrar o alvo diminui.

Os gráficos das Figuras 29 e 30 ilustram o comportamento de cada algoritmo em encontrar o melhor valor de solução em função do tempo para uma instância com 100 nós. As execuções foram realizadas até o tempo de 1800s. Para todos os problemas (MPT, MIMAX, MITOT, MIMAXP E MITOTP), o algoritmo Híbrido encontra os melhores

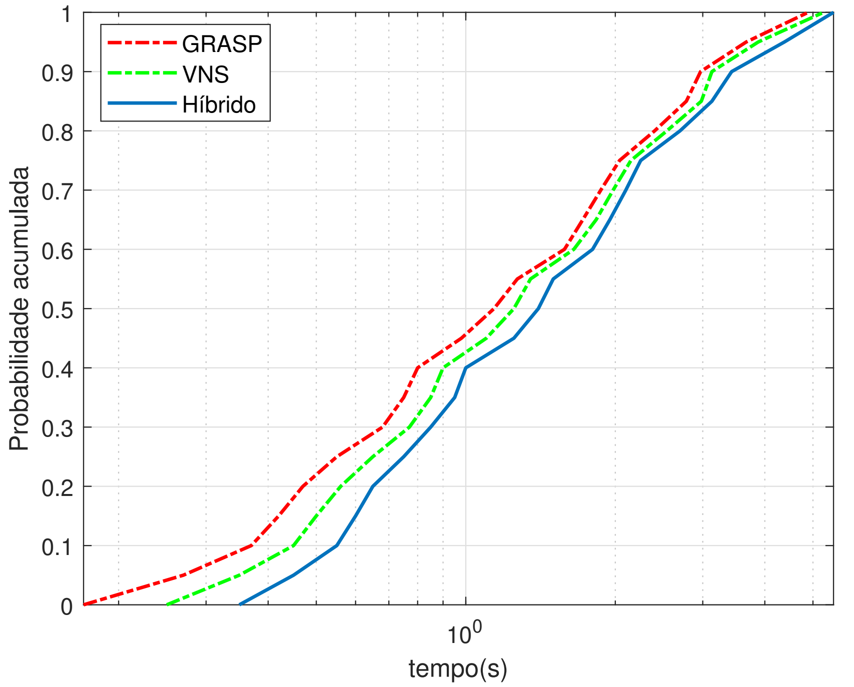


Figura 26 – Gráfico de probabilidade acumulada para EU25 - MITOTP

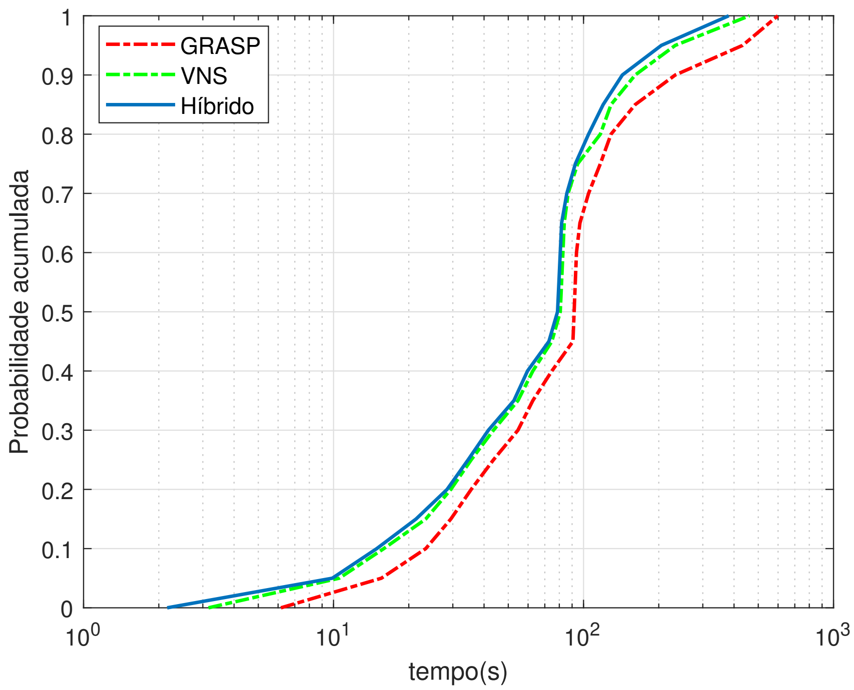


Figura 27 – Gráfico de probabilidade acumulada para EU200 - MITOT

valores de solução mais rápido. Em alguns problemas, o algoritmo VNS também encontra os mesmos valores de solução do Híbrido, mas leva mais tempo.

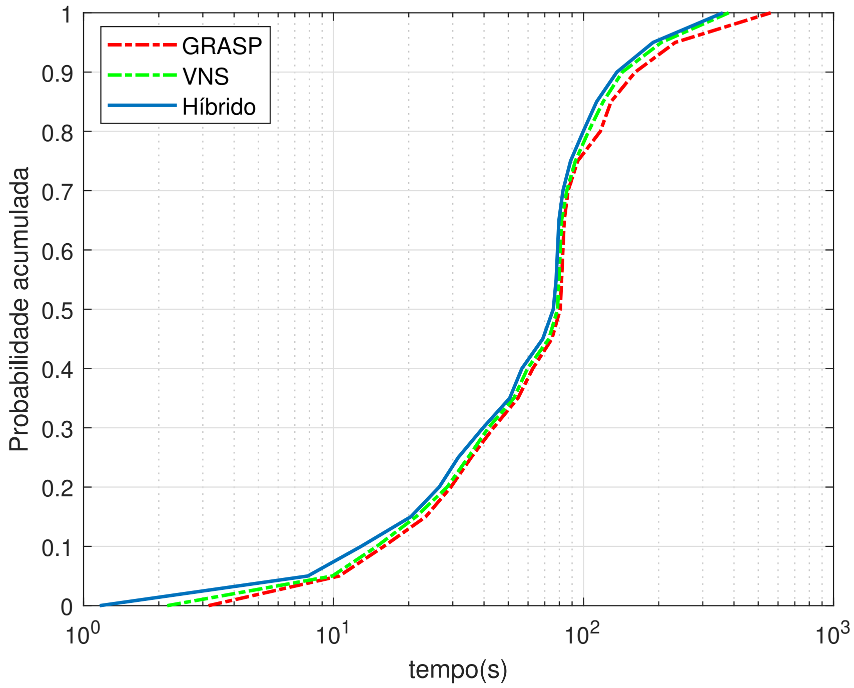


Figura 28 – Gráfico de probabilidade acumulada para EU200 - MITOTP

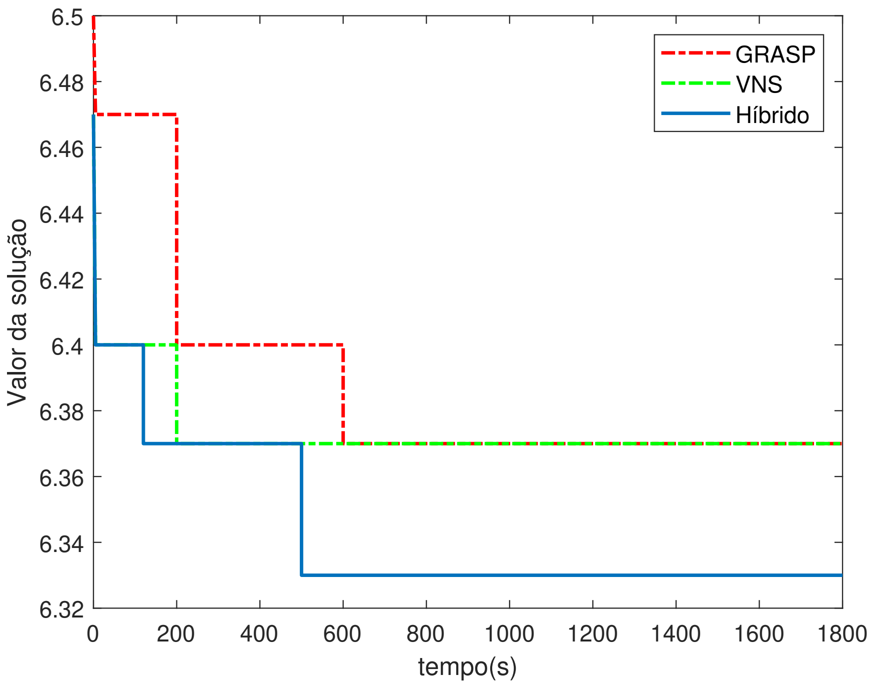


Figura 29 – Gráfico EU100 - MIMAX

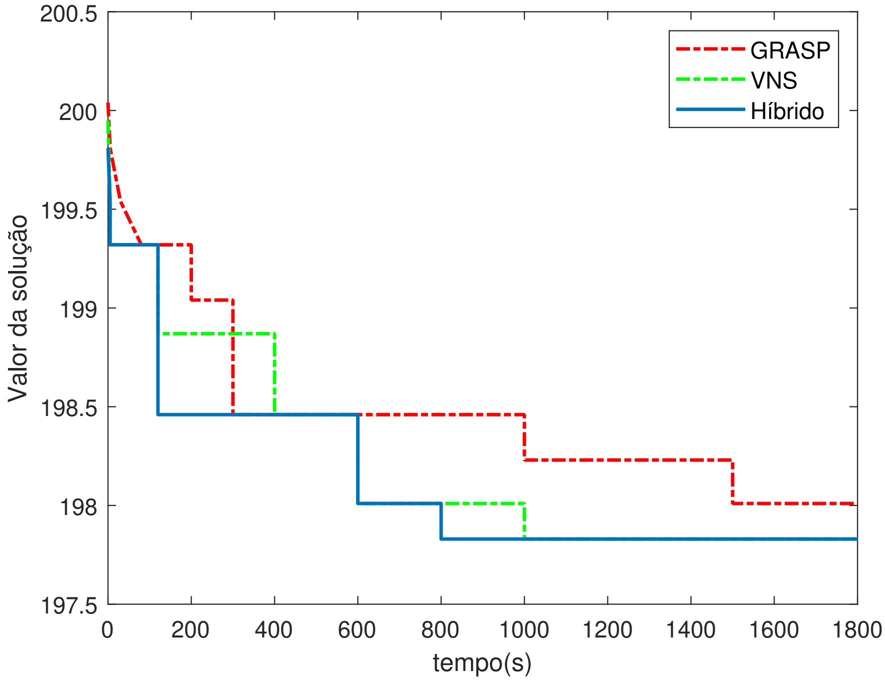


Figura 30 – Gráfico EU100 - MITOT

7 Conclusões e Trabalhos futuros

Neste trabalho foram implementados cinco algoritmos para redução de potência em redes de sensores sem fio quando $k = 1$. Foram os algoritmos gulosos sobre custo de envio e sobre cobertura, a busca local FI/SMEMIT e duas metaheurísticas, GRASP e VND.

Os resultados mostraram que as metaheurísticas GRASP e VND obtiveram bons resultados para potência total comparado ao Guloso sobre custo de envio. Mesmo o GRASP e VND não obtendo os menores valores para a potência, seus valores ficaram bem próximos. Além do mais, a metaheurística VND alcançou os melhores valores de interferência máxima e os valores de potência total ficaram, em média, 0,59% acima do Guloso 1. Então, observa-se que o VND apresenta melhor equilíbrio entre redução de potência e minimização da interferência. Isso torna a metaheurística VND boa na solução do problema em RSSF.

Para $k = 2$, os problemas escolhidos para estudo neste trabalho foram a minimização de potência, interferência máxima e total, e minimização de potência com interferência, simultaneamente para redes com entradas assimétricas, sob restrição de biconexidade.

A primeira proposta de solução destes problemas foi através da metaheurística GRASP. Após, desenvolveu-se a metaheurística VNS e também o algoritmo Híbrido. Primeiramente, as três metaheurísticas foram comparadas entre si através da instância EU. Foi possível observar as características de cada metaheurística desenvolvida, auxiliando na escolha do algoritmo com melhores soluções em menor tempo computacional.

Em seguida, os resultados destes três modelos propostos foram comparados com resultados ótimos encontrados pelo resolvedor CPLEX para as instâncias EU para $|V|$ de 10 a 50 nós. Todos os algoritmos foram comparados em termos de qualidade das soluções obtidas para as cinco variantes do problema (MPT, MIMAX, MIMAXP, MITOT e MITOTP).

As metaheurísticas GRASP e VNS se mostraram promissoras para instâncias bem pequenas, mas como as RSSF são formadas por muitos nós sensores, a metaheurística Híbrida se mostrou mais promissora, pois obteve todas as soluções ótimas para instâncias de pequeno porte. Com isso, foi testado o algoritmo Híbrido para instâncias realísticas do tipo EU, DE e RD, constatando sua eficácia.

Além de comparar as metaheurísticas, também foi realizada uma comparação entre os problemas MPT, MIMAX, MITOT, MIMAXP e MITOTP. Concluiu-se que o problema MPT obtém os menores valores de potências para todos os tamanhos e tipos de instâncias. Os problemas MIMAX e MIMAXP encontram os menores valores de interferência máxima, além de o MIMAXP também obter valores de potência total menores comparados com os valores obtidos pelo MIMAX. Os problemas MITOT e MITOTP encontram os menores valores de interferência total e o MITOTP obtém valores de potência total menores comparados com os valores obtidos pelo MITOT.

Existem diversas formas de estender o trabalho atual e, a partir deste, desenvolver trabalhos futuros. Como ideias, tem-se: estender o tamanho de V, limitar o número de saltos, mobilidade dos nós. Estas novas variações o tornam mais próximo da realidade e das aplicações práticas.

Referências

ABU-AFFASH, A. K.; CARMI, P.; KATZ, M. J. Minimizing total interference in asymmetric sensor networks. *arXiv preprint arXiv:2004.08847*, 2020. Citado na página 24.

AGRAWAL, P.; DAS, G. K. Improved interference in wireless sensor networks. In: SPRINGER. *International Conference on Distributed Computing and Internet Technology*. [S.l.], 2013. p. 92–102. Citado 3 vezes nas páginas 23, 24 e 25.

AIEX, R. M.; RESENDE, M. G.; RIBEIRO, C. C. Ttt plots: a perl program to create time-to-target plots. *Optimization Letters*, Springer, v. 1, n. 4, p. 355–366, 2007. Citado na página 80.

ALTHAUS, E. et al. Power efficient range assignment for symmetric connectivity in static ad hoc wireless networks. *Wireless Networks*, Springer, v. 12, n. 3, p. 287–299, 2006. Citado na página 25.

BASTOS, E. A. Otimização de seções retangulares de concreto armado submetidas à flexo-compressão oblíqua utilizando algoritmos genéticos. *Rio de Janeiro, RJ*, 2004. Citado na página 51.

BILÒ, D.; PROIETTI, G. On the complexity of minimizing interference in ad-hoc and sensor networks. *Theoretical Computer Science*, v. 402, p. 43–55, 2008. Citado na página 36.

BOAVENTURA-NETTO, P. *Grafos: Teoria, Modelos, Algoritmos (in Portuguese). Edgard Blücher, São Paulo*. 2012. Citado na página 16.

BRISE, Y. et al. Interference minimization in asymmetric sensor networks. In: SPRINGER. *International Symposium on Algorithms and Experiments for Sensor Systems, Wireless Networks and Distributed Robotics*. [S.l.], 2014. p. 136–151. Citado na página 23.

BURKHART, M. et al. Does topology control reduce interference? In: *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*. Tokyo: [s.n.], 2004. p. 9–19. Citado 2 vezes nas páginas 22 e 39.

CALINESCU, G. et al. Network lifetime and power assignment in ad hoc wireless networks. In: SPRINGER. *European Symposium on Algorithms*. [S.l.], 2003. p. 114–126. Citado na página 25.

CALINESCU, G.; WAN, P.-J. Range assignment for biconnectivity and k-edge connectivity in wireless ad hoc networks. *Mobile Networks and Applications*, Springer, v. 11, n. 2, p. 121–128, 2006. Citado na página 37.

CHENG, X. et al. Strong minimum energy topology in wireless sensor networks: Np-completeness and heuristics. *IEEE Transactions on mobile computing*, IEEE, v. 2, n. 3, p. 248–256, 2003. Citado na página 21.

CORMEN, T. H. et al. *Introduction to algorithms*. [S.l.]: MIT press, 2009. Citado na página 23.

CORREIA, L. H. et al. Controle de potência de transmissao em protocolos mac para redes de sensores sem fio. In: *Simpósio Brasileiro de Telecomunicaçoes. Campinas. XXII SIMPOSIO BRASILEIRO DE TELECOMUNICACOES*. [S.l.: s.n.], 2005. v. 1, p. 709–714. Citado na página 22.

DAS, A. K. et al. Optimization methods for minimum power bidirectional topology construction in wireless networks with sectored antennas. In: IEEE. *IEEE Global Telecommunications Conference, 2004. GLOBECOM'04.* [S.l.], 2004. v. 6, p. 3962–3968. Citado na página 25.

DU, Y. et al. An energy-efficient and fault-tolerant topology control game algorithm for wireless sensor network. *Electronics*, Multidisciplinary Digital Publishing Institute, v. 8, n. 9, p. 1009, 2019. Citado na página 22.

FARIA, F. C. d. Otimização de seções poligonais de concreto armado sujeitas à flexão composta. 2017. Citado na página 33.

FEO, T. A.; RESENDE, M. G. C. Greedy randomized adaptive search procedures. *Journal of Global Optimization*, v. 6, p. 109–133, 1995. Citado na página 41.

FREITAS, B. D. et al. Programação distribuída para otimização de heurística ils aplicada a problemas do caixeiro viajante. *LINKSCIENCEPLACE-Interdisciplinary Scientific Journal*, v. 1, n. 1, 2014. Citado na página 33.

GENDREAU, M.; POTVIN, J.-Y. et al. *Handbook of metaheuristics*. [S.l.]: Springer, 2010. v. 2. Citado na página 64.

HALLDÓRSSON, M. M.; WATTENHOFER, R. Wireless network algorithmics. In: *Computing and Software Science*. [S.l.]: Springer, 2019. p. 141–160. Citado 4 vezes nas páginas 23, 24, 25 e 35.

HANSEN, P.; MLADENOVIC, N. *Variable neighborhood search methods*. [S.l.]: Springer, 2007. Citado 2 vezes nas páginas 40 e 41.

HÖRNER, D. et al. Resolução do problema das p-medianas não capacitado: comparação de algumas técnicas heurísticas. Florianópolis, SC, 2009. Citado na página 50.

LI, M.; YANG, B. A survey on topology issues in wireless sensor network. In: CITESEER. *ICWN*. [S.l.], 2006. p. 503. Citado na página 32.

LIN, H.-C.; CHEN, W.-Y. An approximation algorithm for the maximum-lifetime data aggregation tree problem in wireless sensor networks. *IEEE Transactions on Wireless Communications*, IEEE, v. 16, n. 6, p. 3787–3798, 2017. Citado na página 30.

LÓPEZ-SÁNCHEZ, A. D.; SÁNCHEZ-ORO, J.; HERNÁNDEZ-DÍAZ, A. G. Grasp and vns for solving the p-next center problem. *Computers & Operations Research*, Elsevier, v. 104, p. 295–303, 2019. Citado na página 59.

LOUREIRO, A. A. et al. Redes de sensores sem fio. In: SN. *Simpósio Brasileiro de Redes de Computadores (SBRC)*. [S.l.], 2003. p. 179–226. Citado 2 vezes nas páginas 18 e 22.

LOUREIRO, A. A. F.; NOGUEIRA, J. M. S.; RUIZ, L. B. Management of wireless sensor networks. In: *Integrated Network Management*. [S.l.: s.n.], 2005. p. 788. Citado na página 28.

LUZ, E. F. P. da; BECCENERI, J. C.; VELHO, H. F. de C. Uma nova metaheurística baseada em algoritmo de colisão de múltiplas partículas. 2008. Citado na página 33.

MARINA, M. K.; DAS, S. R. On-demand multipath distance vector routing in ad hoc networks. In: IEEE. *Proceedings Ninth International Conference on Network Protocols. ICNP 2001*. [S.l.], 2001. p. 14–23. Citado na página 31.

MARINA, M. K.; DAS, S. R. Routing performance in the presence of unidirectional links in multihop wireless networks. In: *Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing*. [S.l.: s.n.], 2002. p. 12–23. Citado na página 37.

MELIÁN, B.; PÉREZ, J. A. M.; VEGA, J. M. M. Metaheurísticas: Una visión global. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, Asociación Española para la Inteligencia Artificial, v. 7, n. 19, p. 0, 2003. Citado 2 vezes nas páginas 33 e 34.

MLADENOVIC, N. A variable neighborhood algorithm-a new metaheuristic for combinatorial optimization. In: *papers presented at Optimization Days*. [S.l.: s.n.], 1995. v. 12. Citado 2 vezes nas páginas 44 e 56.

MOAVENI-NEJAD, K.; LI, X.-Y. Low-interference topology control for wireless ad hoc network. *International Journal of Ad Hoc & Sensor Wireless Networks*, v. 1, p. 41–64, 2005. Citado 3 vezes nas páginas 23, 24 e 25.

MOHANTY, S. K.; UDGATA, S. K. Minimizing the maximum receiver interference in wireless sensor networks using probabilistic interference model. *Engineering Applications of Artificial Intelligence*, Elsevier, v. 91, p. 103563, 2020. Citado na página 23.

MONTEMANNI, R.; GAMBARDELLA, L. M. Exact algorithms for the minimum power symmetric connectivity problem in wireless networks. *Computers & Operations Research*, Elsevier, v. 32, n. 11, p. 2891–2904, 2005. Citado na página 25.

MORAES, R. E.; RIBEIRO, C. C. Power optimization in ad hoc wireless network topology control with biconnectivity requirements. *Computers & Operations Research*, v. 40, n. 12, p. 3188 – 3196, 2013. Citado na página 50.

MORAES, R. E. N. Otimização de potência em redes ad hoc sem fio sob restrições de conexidade. *Tese - Universidade Federal Fluminense*, v. 1, p. 1–142, 2009. Citado 5 vezes nas páginas 47, 53, 54, 55 e 70.

MORAES, R. E. N.; RIBEIRO, C. C.; DUHAMEL, C. Optimal solutions for fault-tolerant topology control in wireless ad hoc networks. *IEEE Transactions on Wireless Communications*, v. 8, p. 5970–5981, 2009. Citado 7 vezes nas páginas 16, 29, 38, 41, 46, 52 e 53.

MORAES, R. E. N. d. M. et al. Joint interference and power minimization for fault-tolerant topology in sensor networks. *Manuscrito*. Citado 3 vezes nas páginas 21, 25 e 46.

NAKAMURA, F. G. et al. Planejamento dinâmico para controle de cobertura e conectividade em redes de sensores sem fio. In: *Workshop de Comunicaçao sem Fio e Computaçao Móvel*. [S.l.: s.n.], 2004. v. 1, p. 182–191. Citado na página 24.

NETO, U. T. Algoritmo de controle de topologia para rede de sensores sem fio que considera o efeito overhearing. Universidade Federal da Bahia. Escola Politécnica/Instituto de Matemática. Citado na página 32.

NETTO, P. O. B. *Grafos: teoria, modelos, algoritmos*. [S.l.]: Editora Blucher, 2003. Citado na página 30.

NIAR, L. I.; HAFFAF, H. Graphical analysis for monitoring in a sensor network (wsn). simulator: Omnet++. In: IEEE. *International Conference on Education and e-Learning Innovations*. [S.l.], 2012. p. 1–6. Citado na página 18.

PANDA, B. S.; SHETTY, D. P. Strong minimum interference topology for wireless sensor networks. In: *Advanced Computing, Networking and Security*. [S.l.]: Springer Berlin / Heidelberg, 2012. v. 7135, p. 366–374. Citado 6 vezes nas páginas 23, 25, 37, 44, 61 e 62.

PINTO, A. J. G. Mecanismo de agregação de dados empregando técnicas paramétricas em redes de sensores. *Rio de Janeiro*, 2004. Citado na página 28.

PRIM, R. C. Shortest connection networks and some generalizations. *The Bell System Technical Journal*, Nokia Bell Labs, v. 36, n. 6, p. 1389–1401, 1957. Citado 2 vezes nas páginas 38 e 39.

RAPPAPORT, T. Wireless communications: Principles and practices. prentice hall, upper saddle river. 2001. Citado na página 35.

SANTI, P. Topology control in wireless ad hoc and sensor networks. *ACM computing surveys (CSUR)*, ACM New York, NY, USA, v. 37, n. 2, p. 164–194, 2005. Citado 2 vezes nas páginas 31 e 37.

SAUSEN, P. S. et al. Gerenciamento integrado de energia e controle de topologia em redes de sensores sem fio. Universidade Federal de Campina Grande, 2008. Citado 2 vezes nas páginas 27 e 28.

SHARMA, A. K. et al. Heuristics for minimizing interference in sensor networks. In: SPRINGER. *International Conference on Distributed Computing and Networking*. [S.l.], 2009. p. 49–54. Citado na página 23.

SHETTY, D. P.; LAKSHMI, M. P. Algorithms for minimizing the receiver interference in a wireless sensor network. In: IEEE. *2016 IEEE Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*. [S.l.], 2016. p. 113–118. Citado 2 vezes nas páginas 23 e 35.

SILVA, J. d. C.; GOMES, E. L.; FRIGIERI, E. P. Controle da topologia de redes de sensores sem fio para economia de energia baseado no algoritmo de prim. *Revista de Tecnológia da Informação e Comunicação*, v. 6, n. 1, p. 9–14, 2016. Citado na página 22.

SILVA, Y. S. et al. Minimização de interferência com redução de potência em redes de sensores sem fio. *LI Simpósio Brasileiro de Pesquisa Operacional*, v. 2, p. 287–299, 2019. Citado na página 25.

SZWARCFITER, J. L. *Teoria computacional de grafos: Os Algoritmos*. [S.l.]: Elsevier Brasil, 2018. Citado na página 29.

TOH, C.-K. Maximum battery life routing to support ubiquitous mobile computing in wireless ad hoc networks. *IEEE communications Magazine*, IEEE, v. 39, n. 6, p. 138–147, 2001. Citado na página 19.

XING, G. et al. Minimum power configuration in wireless sensor networks. In: *Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing*. [S.l.: s.n.], 2005. p. 390–401. Citado na página 18.

YILMAZ, O.; DAGDEVIREN, O.; ERCIYES, K. Interference-aware dynamic algorithms for energy-efficient topology control in wireless ad hoc and sensor networks. *Comput. J.*, v. 54, p. 1398–1411, 2011. Citado na página 37.

ZHANG, L.; ZHU, Q.; LUA, T. Localized topology control for minimizing interference under physical model. *Procedia Engineering*, Elsevier, v. 16, p. 144–150, 2011. Citado 2 vezes nas páginas 24 e 25.

Anexos

ANEXO A – Resultados das formulações exatas - Instâncias euclidianas

Tabela 34 – Resultados obtidos através da programação inteira para formulação MPT

Instância EU	IPBI Potência	Interferência Máxima	Interferência Total
10	1,66	5,47	34,40
15	1,44	5,47	50,53
20	1,45	5,67	67,73
25	1,36	5,67	83,00
30	1,32	5,87	98,87
35	1,26	6,27	113,55
40	1,25	5,75	121,75
45	1,23	6,00	136,00
50	1,21	6,25	153,75

Tabela 35 – Resultados obtidos através da programação inteira para formulação MIMAX

Instância EU	IPBI Potência	Interferência Máxima	Interferência Total
10	2,38	5,07	41,53
15	2,05	4,93	61,47
20	2,06	5,07	84,33
25	2,05	5,33	112,67
30	1,85	5,00	126,87
35	1,94	5,09	151,73
40	1,93	5,50	147,00
45	1,64	5,00	176,00
50	1,78	5,00	223,50

Tabela 36 – Resultados obtidos através da programação inteira para formulação MIMAXP

Instância EU	IPBI Potência	Interferência Máxima	Interferência Total
10	1,69	5,07	34,00
15	1,48	4,93	50,20
20	1,49	5,07	68,00
25	1,40	5,33	83,53
30	1,35	5,00	99,47
35	1,33	5,00	115,00
40	1,15	5,50	123,50
45	1,18	5,00	139,50
50	1,16	5,00	152,3

Tabela 37 – Resultados obtidos através da programação inteira para formulação MITOT

Instância EU	IPBI Potência	Interferência Máxima	Interferência Total
10	1,78	5,27	33,00
15	1,52	5,40	48,60
20	1,53	5,53	65,60
25	1,45	5,80	80,87
30	1,36	5,80	97,53
35	1,27	5,85	104,43
40	1,23	5,50	118,25
45	1,12	4,00	125,00
50	–	–	–

Tabela 38 – Resultados obtidos através da programação inteira para formulação MITOTP

Instância EU	IPBI Potência	Interferência Máxima	Interferência Total
10	1,73	5,33	33,00
15	1,50	5,20	48,60
20	1,53	5,60	65,60
25	1,44	5,73	80,87
30	1,34	5,73	96,20
35	1,29	6,00	109,18
40	1,20	5,75	122,00
45	1,12	5,00	125,00
50	–	–	–

Apêndices

APÊNDICE A – Resultados das instâncias EU, DE e RD para MPT, MIMAX, MITOT, MIMAXP e MITOTP - $|V|$ de 25 a 800 nós

Tabela 39 – Resultados das instâncias EU, DE e RD para minimização de potência (MPT)

	Potência			Interf. Máx.			Interf. Tot.		
|V|	EU	DE	RD	EU	DE	RD	EU	DE	RD
25	1,36	5,46	5,52	5,67	6,00	5,67	83,00	115,25	101,14
50	1,27	10,21	8,45	6,25	7,33	6,27	153,75	205,59	183,64
100	1,35	19,42	11,73	7,27	7,47	7,27	207,36	236,78	220,64
200	1,69	37,52	17,54	7,33	8,67	7,47	216,35	388,29	232,75
400	2,81	74,12	25,13	7,83	10,53	7,83	232,68	565,67	242,37
800	5,04	108,63	39,47	8,00	10,83	8,27	312,37	702,89	342,75

Tabela 40 – Resultados das instâncias EU, DE e RD para minimização de interferência máxima (MIMAX)

	Potência			Interf. Máx.			Interf. Tot.		
|V|	EU	DE	RD	EU	DE	RD	EU	DE	RD
25	2,05	5,68	5,91	5,33	5,47	5,47	112,67	110,52	103,65
50	1,78	10,47	8,87	5,00	6,93	6,07	223,50	202,75	188,76
100	1,95	19,76	12,27	6,33	7,27	7,00	214,39	233,34	222,92
200	2,17	37,94	18,32	7,27	8,00	7,33	223,64	382,28	233,84
400	3,02	75,12	25,95	7,27	10,00	7,47	239,35	563,86	246,27
800	5,39	113,23	42,64	7,83	10,27	7,83	309,68	698,42	338,27

Tabela 41 – Resultados das instâncias EU, DE e RD para minimização de interferência total (MITOT)

	Potência			Interf. Máx.			Interf. Tot.		
|V|	EU	DE	RD	EU	DE	RD	EU	DE	RD
25	1,45	5,59	5,72	5,80	5,80	5,67	80,87	108,75	98,35
50	1,38	10,35	8,63	6,07	7,27	6,33	182,50	200,62	182,98
100	1,47	19,58	11,97	6,47	7,33	7,27	201,35	231,13	217,25
200	1,80	37,71	17,93	7,27	8,53	7,47	214,69	379,89	229,37
400	2,87	74,33	25,43	7,33	10,53	7,67	227,05	562,04	243,85
800	5,08	112,89	41,89	7,93	10,47	8,00	307,08	697,63	337,92

Tabela 42 – Resultados das instâncias EU, DE e RD para minimização de interferência máxima e potência (MIMAXP)

	Potência			Interf. Máx.			Interf. Tot.		
|V|	EU	DE	RD	EU	DE	RD	EU	DE	RD
25	1,40	5,51	5,57	5,33	5,67	5,33	83,53	109,75	102,23
50	1,32	10,24	8,47	5,00	6,93	6,07	152,30	203,12	184,46
100	1,39	19,48	11,80	6,33	7,27	7,00	213,58	230,34	218,69
200	1,75	37,58	17,64	7,07	8,00	7,27	221,82	381,94	231,72
400	2,82	74,17	25,19	7,27	10,00	7,33	233,68	563,52	242,52
800	5,05	109,25	40,82	7,83	10,33	7,83	309,12	698,07	337,78

Tabela 43 – Resultados das instâncias EU, DE e RD para minimização de interferência total e potência (MITOTP)

	Potência			Interf. Máx.			Interf. Tot.		
|V|	EU	DE	RD	EU	DE	RD	EU	DE	RD
25	1,44	5,55	5,65	5,73	5,83	5,53	80,87	106,82	96,65
50	1,35	10,29	8,55	6,00	7,00	6,27	150,78	199,23	181,74
100	1,42	19,52	11,89	6,47	7,33	7,27	200,03	229,47	215,32
200	1,77	37,64	17,77	7,27	8,27	7,47	211,56	377,59	227,56
400	2,84	74,23	25,27	7,33	10,27	7,67	228,46	559,98	240,78
800	5,06	109,64	41,63	7,93	10,47	8,00	306,78	695,33	336,98

APÊNDICE B – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrido para instâncias EU200, EU 400 e EU800 para o problema MPT

Tabela 44 – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrido para a instância EU200 para o problema MPT

	Tempo (s)					
Tipo	5	30	120	300	600	1800
GRASP	1,7384	1,7368	1,7342	1,7313	1,7304	1,7297
BVNS	1,7372	1,7356	1,7337	1,7308	1,7295	**1,7289**
HÍBRIDO	**1,7369**	**1,7348**	**1,7322**	**1,7303**	**1,7289**	**1,7289**

Tabela 45 – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrido para a instância EU400 para o problema MPT

	Tempo (s)					
Tipo	5	30	120	300	600	1800
GRASP	3,0764	3,0402	3,0348	3,0234	2,9856	2,9467
BVNS	3,0823	3,0386	3,0312	3,0145	2,9726	2,9435
HÍBRIDO	**3,0576**	**3,0378**	**3,0246**	**2,9825**	**2,9549**	**2,8802**

Tabela 46 – Evolução dos resultados obtidos pelas metaheurísticas GRASP, VNS e Híbrido para a instância EU800 para o problema MPT

	Tempo (s)					
Tipo	5	30	120	300	600	1800
GRASP	5,1195	5,1183	5,1172	5,1097	5,1012	5,0945
BVNS	5,1183	5,1158	5,1133	5,1068	5,0979	5,0922
HÍBRIDO	**5,1083**	**5,1034**	**5,0946**	**5,0895**	**5,0834**	**5,0794**

APÊNDICE C – Gráficos de probabilidade acumulada para as metaheurísticas GRASP, VNS e Híbrido para instâncias EU25 e EU200

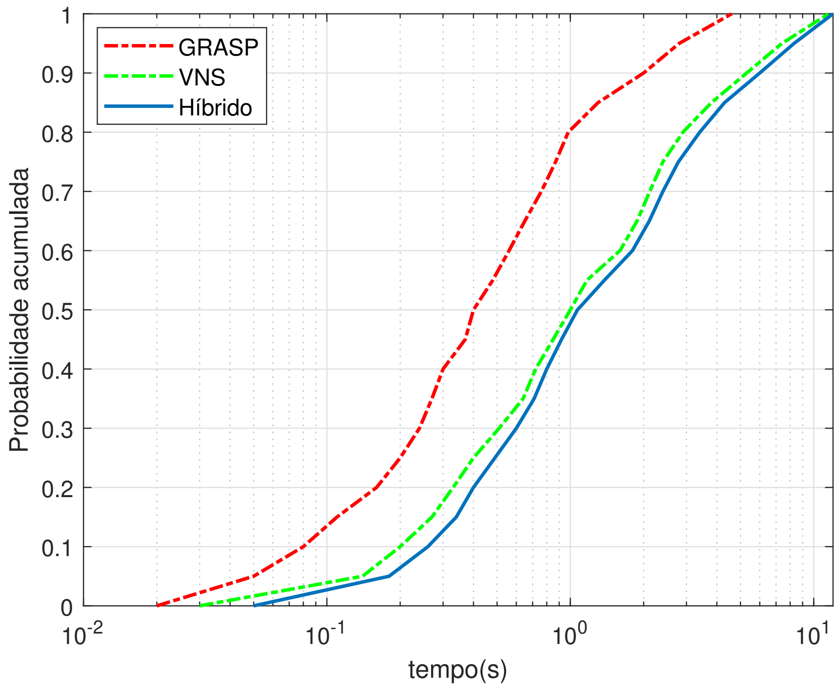


Figura 31 – Gráfico de probabilidade acumulada para EU25 - MPT

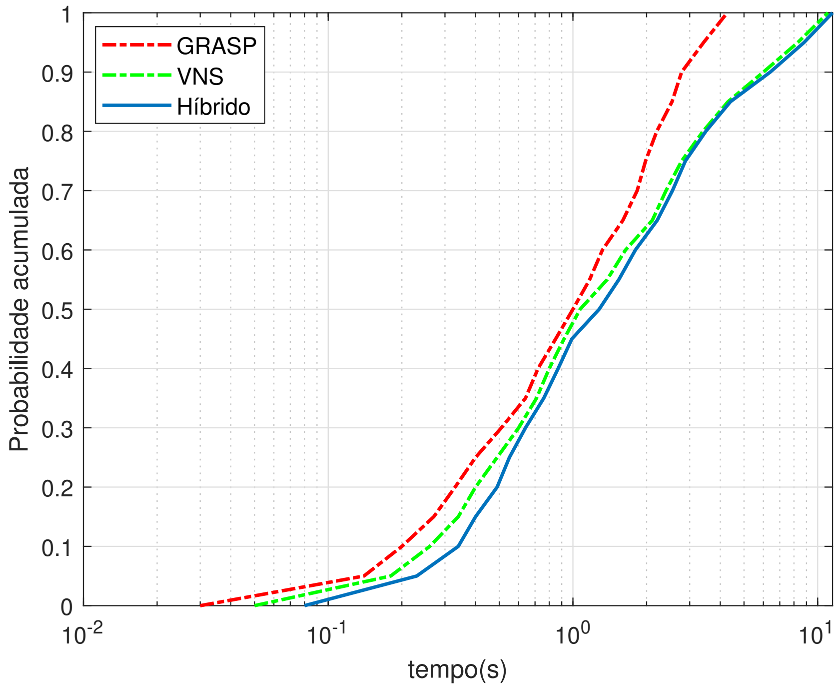


Figura 32 – Gráfico de probabilidade acumulada para EU25 - MIMAX

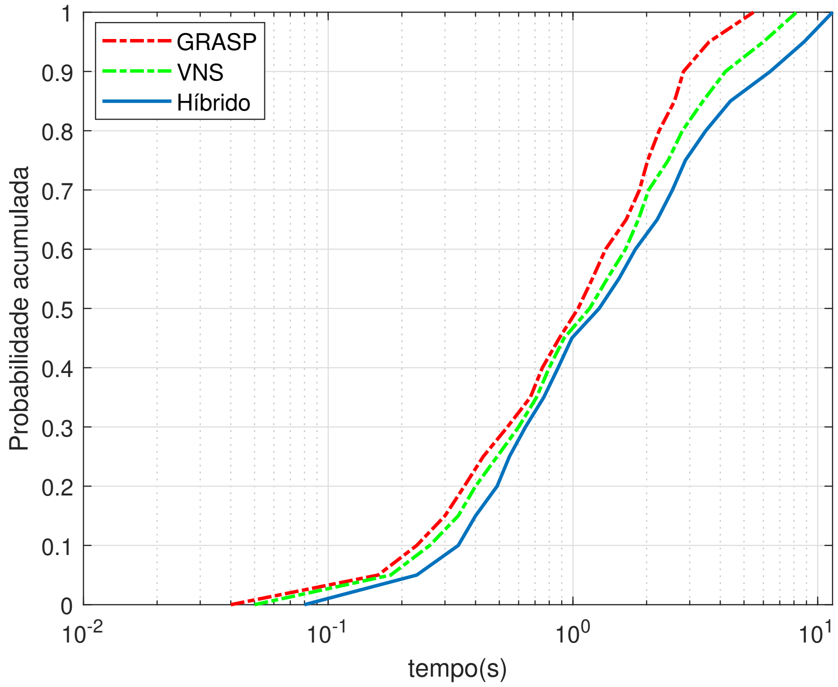


Figura 33 – Gráfico de probabilidade acumulada para EU25 - MIMAXP

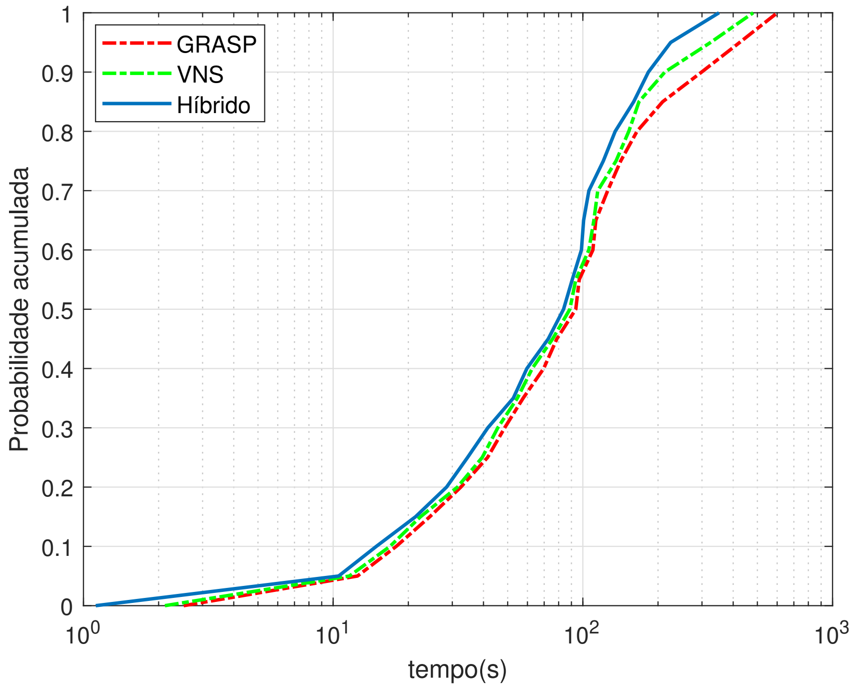


Figura 34 – Gráfico de probabilidade acumulada para EU200 - MPT

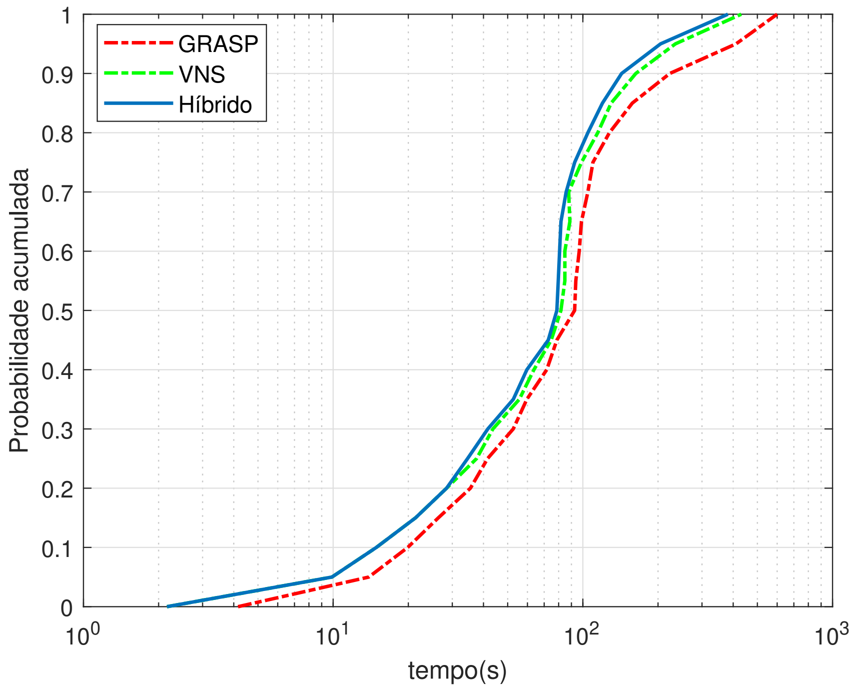


Figura 35 – Gráfico de probabilidade acumulada para EU200 - MIMAX

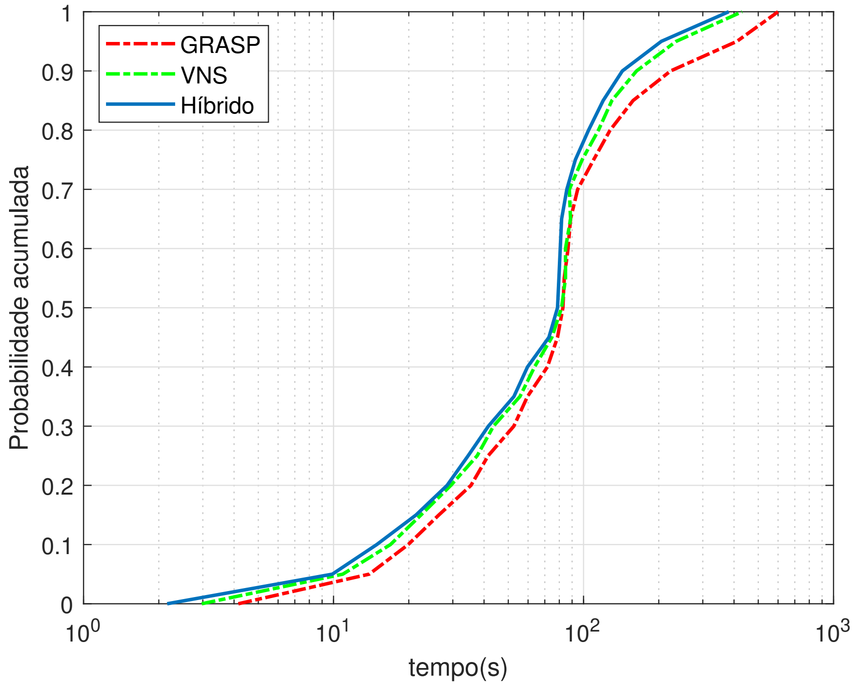


Figura 36 – Gráfico de probabilidade acumulada para EU200 - MIMAXP

APÊNDICE D – Gráficos para as metaheurísticas GRASP, VNS e Híbrido para instância EU100

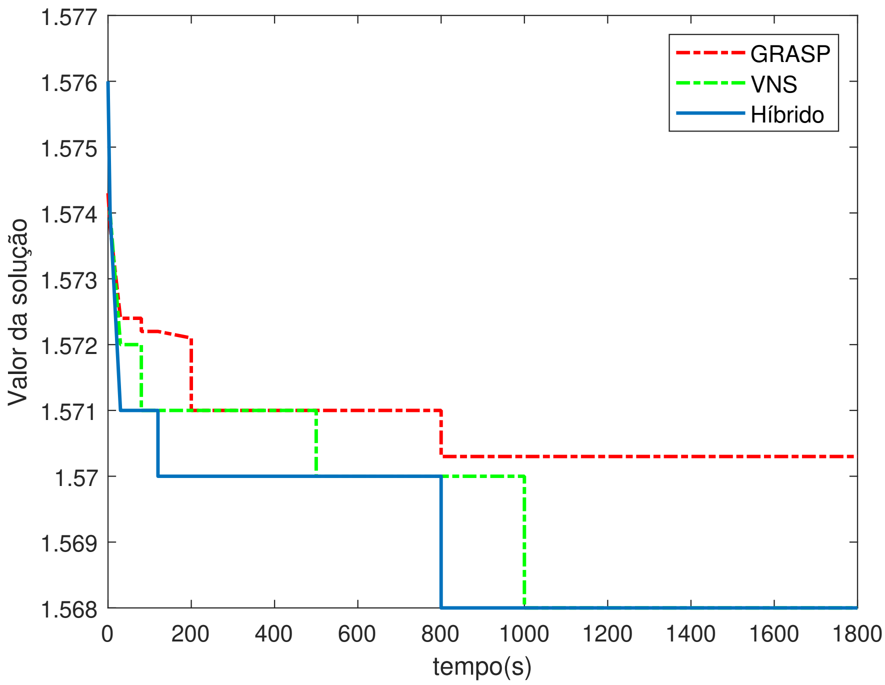


Figura 37 – Gráfico EU100 - MPT

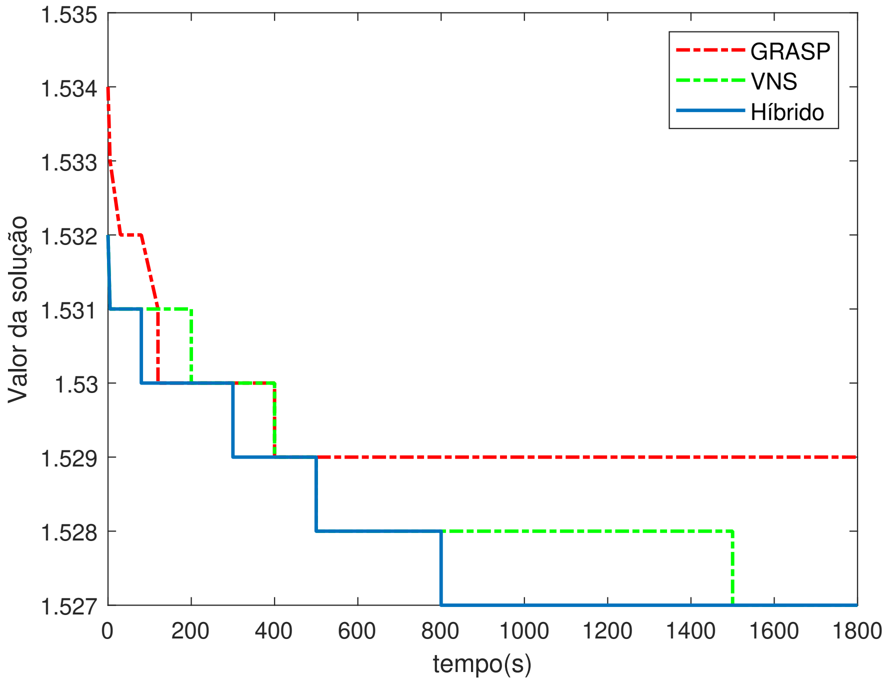


Figura 38 – Gráfico EU100 - MIMAXP

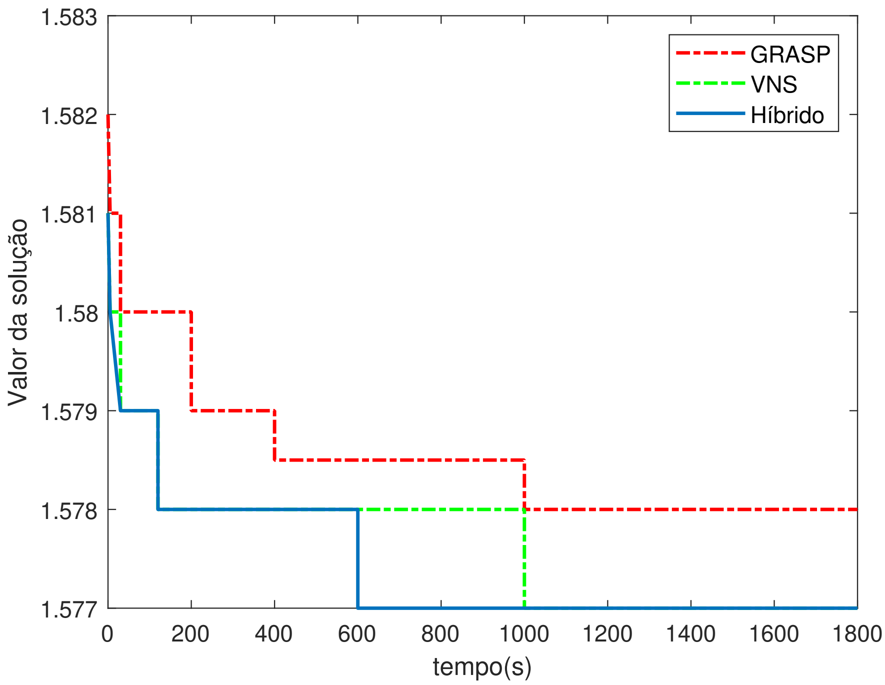


Figura 39 – Gráfico EU100 - MITOTP